

Quantitative Timed Simulation Functions and Refinement Metrics for Timed Systems *

Krishnendu Chatterjee¹ and Vinayak S. Prabhu²

¹ IST Austria (Institute of Science and Technology Austria); ²FEUP, University of Porto;
 Krishnendu.Chatterjee@ist.ac.at, vinayak@eecs.berkeley.edu

Abstract. We introduce quantitative timed refinement metrics and quantitative timed simulation functions, incorporating zenoness checks, for timed systems. These functions assign positive real numbers between zero and infinity which quantify the *timing mismatches* in between two timed systems, amongst non-zero runs. We quantify timing mismatches in three ways: (1) the maximum timing mismatch that can arise, (2) the “steady-state” maximum timing mismatches, where initial transient timing mismatches are ignored; and (3) the (long-run) average timing mismatches amongst two systems. These three kinds of mismatches constitute three important types of timing differences. Our event times are the *global times*, measured from the start of the system execution, not just the time durations of individual steps. We present algorithms over timed automata for computing the three quantitative simulation functions to within any desired degree of accuracy. In order to compute the values of the quantitative simulation functions, we use a game theoretic formulation. We introduce two new kinds of objectives for two player games on finite state game graphs: (1) eventual debit-sum level objectives, and (2) average debit-sum level objectives. We present algorithms for computing the optimal values for these objectives for player 1, and then use these algorithms to compute the values of the quantitative timed simulation functions.

1 Introduction

Theories of system approximation for continuous systems are used for analyzing systems that differ to a small extent, as opposed to the traditional boolean yes/no view of system refinement for discrete systems. These theories are necessary as formal models are only approximations of the real world, and are subject to estimation and modelling errors. Approximation theories have been traditionally developed for continuous control systems [ASG01] and more recently for linear and non-linear systems [GJP08; GPT10; Pol+10], timed systems [HMP05], labeled Markov Processes [Des+04], probabilistic automata [Bre+03], quantitative transition systems [AFS09], and software systems [CGL12].

Timed and hybrid systems model the evolution of system outputs as well as the timing aspects related to the system evolution. In this work we develop a theory of system approximation for timed systems by quantifying the *timing differences* between corresponding system events. We first generalize timed refinement relations to metrics on timed systems that quantitatively estimate the closeness of two systems. Given a timed model T_s denoting the abstract specification model, and a model T_r denoting the concrete refined implementation of T_s , we assign a positive real number between zero and infinity to the pair (T_r, T_s) which denotes the quantitative refinement distance between T_r and T_s . Given a trace tr_r of T_r , and a trace tr_s of T_s , we define various distances

* This work has been financially supported in part by the European Commission FP7-ICT Cognitive Systems, Interaction, and Robotics under the contract # 270180 (NOPTILUS); by Fundação para Ciência e Tecnologia under project PTDC/EEA-CRO/104901/2008 (Modeling and control of Networked vehicle systems in persistent autonomous operations); by Austrian Science Fund (FWF) Grant No P 23499-N23 on Modern Graph Algorithmic Techniques in Formal Verification; FWF NFN Grant No S11407-N23 (RiSE); ERC Start grant (279307: Graph Games); and the Microsoft faculty fellows award



Fig. 1. Two timed automata $\mathcal{T}_1, \mathcal{T}_2$.

between the two traces, e.g., the distance being ∞ if the untimed trace sequences differ, and being the supremum of the differences of the matching timepoints for matching events otherwise. Our event times are the *global times*, measured from the start of the system execution, not just the time durations of individual steps. The distance between the systems T_r and T_s is taken to be the supremum of closest matching trace differences from the initial states.

Timed trace inclusion is undecidable on timed automata [AD94], thus timed refinement is conservatively estimated using *timed simulation relations* [Cer92]. Simulation relations take a branching time view, unlike the linear view of refinement relations, and can be defined using two player *games*. We generalize timed simulation relations to quantitative timed simulation functions, and define the values of quantitative timed simulation functions as the real-valued outcome of games played on the corresponding timed graphs.

Zeno runs where time converges is an artifact present in models of timed systems due to model imperfections; such runs are obviously absent in the physical systems which our timed models are meant to represent. We thus exclude Zeno runs in our computation of quantitative timed refinement and quantitative timed simulation relations.

We define three illustrative quantitative timed simulation functions which measure three important system differences. The *maximum time difference* quantitative simulation function denotes the maximum time discrepancy that can arise amongst matching transitions. The *eventual maximum time difference* quantitative simulation function denotes the eventual maximum time discrepancy that arises (ignoring finite time trace prefix discrepancies) amongst matching transitions. This corresponds to the “steady-state” difference between systems, ignoring transient behavior. The (*long-run*) *average time difference* quantitative simulation function denotes the average time discrepancy amongst matching transitions. This function measures the long-run average time discrepancies, per transition, amongst two timed systems. Ideally, we want all three simulation functions to be as small as possible between the specification and the implementation systems, but minimizing one may lead to increase in values for others. Thus, all three simulation functions give important information about systems. We illustrate the various quantitative timed simulation functions via examples.

Example 1 (Maximum Time Difference). Consider the two timed automata \mathcal{T}_1 and \mathcal{T}_2 in Figure 1. The locations are labelled with the observations. The starting location of each automaton is the one labelled with the observation a , and the starting value of the clock x is 0. Let us look at the value of the *maximum time difference* quantitative timed simulation function $\mathcal{S}_{\text{MaxDiff}}$ for the state pair $(\langle a, x = 0 \rangle^{\mathcal{T}_1}, \langle a, x = 0 \rangle^{\mathcal{T}_2})$. The value is (1) infinity if every transition from the state in \mathcal{T}_1 cannot be matched by a transition from the matching state in \mathcal{T}_2 (and similarly for following steps), that is the state of \mathcal{T}_1 *time-abstract simulates* the state of \mathcal{T}_2 ; (2) the maximum time difference between matching transitions of \mathcal{T}_1 and \mathcal{T}_2 otherwise, amongst *time-divergent* runs. For

the two timed automata in Figure 1, it can be checked that $\langle a, x = 0 \rangle^{\mathcal{T}_1}$ time-abstract simulates $\langle a, x = 0 \rangle^{\mathcal{T}_2}$, and that the maximum time difference between matching transitions is 9 time units, (e.g. between the paths $\langle a, x = 0 \rangle^{\mathcal{T}_1} \xrightarrow{10} \langle b, x = 0 \rangle^{\mathcal{T}_1} \xrightarrow{0} \langle c, x = 0 \rangle^{\mathcal{T}_1} \xrightarrow{5} \langle c, x = 0 \rangle^{\mathcal{T}_1} \xrightarrow{5} \dots$ and $\langle a, x = 0 \rangle^{\mathcal{T}_2} \xrightarrow{1} \langle b, x = 0 \rangle^{\mathcal{T}_2} \xrightarrow{9} \langle c, x = 0 \rangle^{\mathcal{T}_2} \xrightarrow{5} \langle c, x = 0 \rangle^{\mathcal{T}_2} \xrightarrow{5} \dots$). \square

Example 2 (Global Event Times). Consider the two timed automata in Figure 2. The value of



Fig. 2. Two timed automata $\mathcal{T}_3, \mathcal{T}_4$.

the maximum time difference quantitative timed simulation function $\mathcal{S}_{\text{MaxDiff}}$ for the state pair $(\langle a, x = 0 \rangle^{\mathcal{T}_3}, \langle a, x = 0 \rangle^{\mathcal{T}_4})$ is ∞ , since timing mismatch corresponding to the n -th transition is n (the n -th transition in \mathcal{T}_3 occurs at global time n , the n -th transition in \mathcal{T}_4 occurs at global time $2 \cdot n$). We depict the timelines in Figure 3. \square

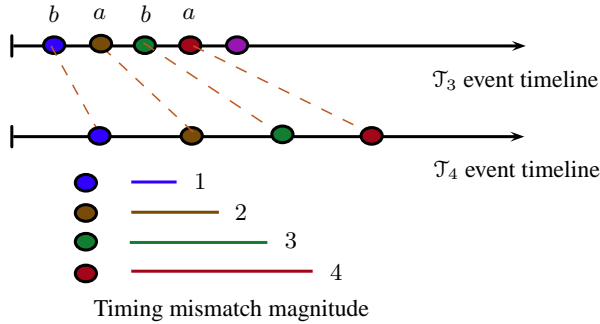


Fig. 3. Timeline of $\mathcal{T}_3, \mathcal{T}_4$ events.

Example 3 (Eventual Maximum Time Difference). Consider the two timed automata \mathcal{T}_1 and \mathcal{T}_2 in Figure 1. Let us look at the value of the *eventual maximum time difference* quantitative timed simulation function $\mathcal{S}_{\text{LimMaxDiff}}$ for the state pair $(\langle a, x = 0 \rangle^{\mathcal{T}_1}, \langle a, x = 0 \rangle^{\mathcal{T}_2})$. The value is (1) infinity if every transition from the state in \mathcal{T}_1 cannot be matched by a transition from the matching state in \mathcal{T}_2 (and similarly for following steps), that is the state of \mathcal{T}_1 *time-abstract simulates* the state of \mathcal{T}_2 ; (2) the *eventual* maximum time difference between matching transitions of \mathcal{T}_1 and \mathcal{T}_2 otherwise (ignoring the time differences amongst finite trace prefixes), amongst *time-divergent* runs. In the automata $\mathcal{T}_1, \mathcal{T}_2$, there is a time mismatch only at the transitions from a , and this transition can only occur before time 10. Once the executions reach the location c , the automaton \mathcal{T}_2 is able to match the transitions of \mathcal{T}_1 at the exact times, with zero time discrepancy. Thus, $\mathcal{S}_{\text{LimMaxDiff}}$ denotes the “steady-state” time discrepancy between $\mathcal{T}_1, \mathcal{T}_2$, and this value is zero for the state pair $(\langle a, x = 0 \rangle^{\mathcal{T}_1}, \langle a, x = 0 \rangle^{\mathcal{T}_2})$, in contrast to the value of 9 for $\mathcal{S}_{\text{MaxDiff}}$ for the state pair. Note that we ignore time-discrepancies for finite *time* (by discarding Zeno runs), not just finite trace prefixes.

If we ignore only finite trace prefixes, then we would have obtained a value of 9, as \mathcal{T}_1 can loop on the location b by preventing time from progressing (note that the clock x is not reset on the b loop transition). \square

Example 4 (Eventual Maximum Time Difference). Consider the two timed automata \mathcal{T}_5 and \mathcal{T}_6



Fig. 4. Two timed automata $\mathcal{T}_5, \mathcal{T}_6$.

in Figure 4. Let us look at the value of the *eventual maximum time difference* quantitative timed simulation function $\mathcal{S}_{\text{LimMaxDiff}}$ for the state pair $(\langle a, x = 0 \rangle^{\mathcal{T}_5}, \langle a, x = 0 \rangle^{\mathcal{T}_6})$. In this case, a time difference of 9 occurs infinitely often in *time-divergent* runs, (e.g. between the paths $\langle a, x = 0 \rangle^{\mathcal{T}_5} \xrightarrow{10} \langle b, x = 0 \rangle^{\mathcal{T}_5} \xrightarrow{0} \langle c, x = 0 \rangle^{\mathcal{T}_5} \xrightarrow{5} \langle a, x = 0 \rangle^{\mathcal{T}_5} \xrightarrow{10} \dots$ and $\langle a, x = 0 \rangle^{\mathcal{T}_6} \xrightarrow{1} \langle b, x = 0 \rangle^{\mathcal{T}_6} \xrightarrow{9} \langle c, x = 0 \rangle^{\mathcal{T}_6} \xrightarrow{5} \langle a, x = 0 \rangle^{\mathcal{T}_6} \xrightarrow{1} \dots$). The maximum time difference of 9 time units arises when taking the transitions from the a -labelled states. Thus, the value of $\mathcal{S}_{\text{LimMaxDiff}}$ for the state pair $(\langle a, x = 0 \rangle^{\mathcal{T}_5}, \langle a, x = 0 \rangle^{\mathcal{T}_6})$ is 9. It can be checked that in this case, the value of $\mathcal{S}_{\text{MaxDiff}}$ for the state pair is also 9. \square

Example 5 (Average Time Difference). Consider the two timed automata \mathcal{T}_5 and \mathcal{T}_6 in Figure 4. Let us look at the value of the (long-run) *average time difference* quantitative timed simulation function $\mathcal{S}_{\text{AvgDiff}}$ for the state pair $(\langle a, x = 0 \rangle^{\mathcal{T}_5}, \langle a, x = 0 \rangle^{\mathcal{T}_6})$. As usual, for the value to be finite, we require time-abstract simulation. If time-abstract simulation holds, we take the average with respect to the number of transitions (over non-Zeno runs). For the state pair, a time difference of 9 occurs infinitely often, but this difference occurs in only one-third of the transitions (the transitions from a locations). For the transitions from b and c , the time discrepancy is zero. Thus, the value for $\mathcal{S}_{\text{AvgDiff}}$ is $\frac{9+0+0}{3} = 3$. \square

To compute the values of the three simulation functions, we use the framework of turn based games on finite-state game graphs. We introduce two new game theoretic objectives (these objectives are required for computing two of the quantitative simulation functions) on these game graphs, namely, *eventual debit-sum level* and *average debit-sum level* objectives, and present novel solutions for both. We need to consider the sums of the weights encountered as in our quantitative simulation functions, the global time is the sum of the time durations of all the preceding transitions.

Eventual debit-sum level and average debit-sum level games are also interesting on their own. We next illustrate average debit-sum level games. These games are played on two-player turn based game graphs. Each transition in the game graph incurs a cost (denoted by a negative weight), or a reward (denoted by a positive weight). These costs can be viewed as monetary losses, or monetary gains. The debit-sum level at a stage in the game denotes the absolute value of the monetary balance, if the balance is negative (the balance is the sum of all the positive and negative costs and rewards).

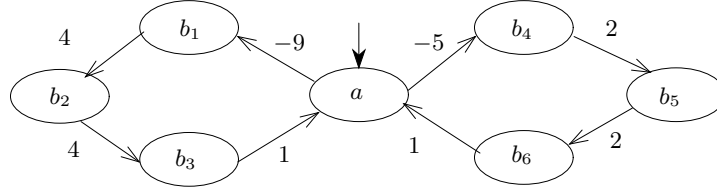


Fig. 5. Debit sum-level game

The objective of player 1 is to have the lowest possible average debit-sum level. These games are also applicable, for instance, in financial markets, where banks have to take overnight loans from the Federal Reserve loan windows in case of need (these loans need to be renewed each day the loan is not repaid). It is in the banks interests to minimize the average of the loan amount per day.

Example 6 (Debit Sum-Level Turn Based Games). Consider the turn based game depicted in Figure 5. The only player-1 location is a , the other locations are player-2 locations. The numbers on the edges denote the costs or rewards that player-1 gets when that transition is taken. Positive weights denotes rewards, and negative weights denotes costs. Viewing the weights as monetary transactions, and starting with a monetary balance of zero at a , if player 1 loops around the left loop, then the trace, together with the monetary balances is: $((a, 0) (b_1, -9) (b_2, -5) (b_3, -1))^\omega$, where the numbers denote the accumulated balances during the run of the play. The average negative balance, *i.e.*, the average debit-sum level (per unit location visit), is $\frac{0+9+5+1}{4} = \frac{15}{4}$. If player 1 loops around the right loop, then the trace, together with the balances is: $((a, 0) (b_4, -5) (b_5, -3) (b_6, -1))^\omega$. The average negative balance is $\frac{0+5+3+1}{4} = \frac{9}{4}$. Thus the optimum average debit sum-level value for player 1 is $9/4$, and the optimum strategy is to loop around the right-hand side, where it needs to borrow less, on average. \square

Our Contributions. Our main contributions in the present work are as follows.

- ★ We define three quantitative refinement metrics incorporating Zenoness conditions semantically, that is our refinement metrics ignore artificial Zeno runs present in systems due to modelling artifacts. We also show that these quantitative functions are actually (directed) metrics.
- ★ We define quantitative timed simulation functions corresponding to the refinement metrics using a game theoretic formulation. These quantitative simulation functions also incorporate Zenoness conditions for obtaining physically meaningful system differences. As far we know, this is the first work which handles Zeno runs when computing simulation functions.
- ★ We present *decision procedures* for computing all the defined quantitative timed simulation functions to within any desired degree of accuracy for *any* given timed automaton.
- ★ We introduce new game theoretic objectives on finite-state game graphs, namely, eventual debit-sum level objectives and average debit-sum level objectives, and present novel solutions for both on finite-state turn based games. These new objectives are required in the computation of the defined quantitative simulation functions.

We have considered the (more challenging) framework of global event times in our quantitative simulation functions. Our solution framework is also applicable where the mismatches are only with respect to transition *durations* (simple algorithms are applicable in this case). Our algorithms can easily be generalized to consider quantitative simulation functions in which an observation σ is allowed to match a different observation σ' , but with some matching penalty in case $\sigma \neq \sigma'$ (the penalty being in addition to the timing mismatch of σ, σ'). Thus, our algorithms apply to the

computation of quantitative simulation functions which consider the *Skorokhod metric* [JS03] over mismatches.

Related Work. The most relevant related work is the recent work on the theory of *approximate bisimulation* for continuous and switched systems [GJP08; GPT10], and [QFD11]. The approximations in [GJP08; GPT10] are with respect to the real-valued system outputs, and not with respect to the *times* during which the values are output. The simulation relations are constrained to match values at equivalent sample points, thus there is no mechanism to judge the time discrepancies. The work in [QFD11] presents similarity relations where the approximations are with respect to time as well as output values. Computation of similarity relations is reduced to solving a derived game, however, no decidability results are presented for solving these derived games. For timed systems, the work in [HMP05] presented maximum time difference quantitative timed simulation functions, however, Zeno issues were ignored. Our solutions for the new objectives on finite-state game graphs builds on previous work on mean payoff parity games, multi-dimensional mean payoff, and energy games [Bou+11; CD10; Cha+10; Cha10; CHJ05]. The new game objectives presented in the present work, that are required for the quantitative timed simulation functions, were previously unstudied, and require new ideas in their solutions.

2 Quantitative Timed Trace Difference and Refinement Metrics

We define *quantitative* refinement functions on timed systems. These functions allow approximate matching of timed traces and generalize timed and untimed refinement relations.

Timed Transition System. A *timed transition system* (TTS) is a tuple $A = \langle S, \Sigma, \rightarrow, \mu, S_0 \rangle$ where

- S is the set of states.
- Σ is a set of atomic propositions (the observations).
- $\rightarrow \subseteq S \times \mathbb{R}^+ \times S$ is the transition relation such that for all $s \in S$ there exists at least one $s' \in S$ such that for some Δ , we have $s \xrightarrow{\Delta} s'$.
- $\mu : S \mapsto 2^\Sigma$ is the observation map which assigns a truth value to atomic propositions true in a state.
- $S_0 \subseteq S$ is the set of initial states.

We write $s \xrightarrow{t} s'$ if $(s, t, s') \in \rightarrow$. A *state trajectory* is an infinite sequence $s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} \dots$, where for each $j \geq 0$, we have $s_j \xrightarrow{t_j} s_{j+1}$. The state trajectory is *initialized* if $s_0 \in S_0$ is an initial state. A state trajectory $s_0 \xrightarrow{t_0} s_1 \dots$ induces a *trace* given by the observation sequence $\mu(s_0) \xrightarrow{t_0} \mu(s_1) \xrightarrow{t_1} \dots$. To emphasize the initial state, we say s_0 -trace for a trace induced by a state trajectory starting from s_0 . A trace is initialized if it is induced by an initialized state trajectory. Given a trace \mathbf{tr} induced by a state trajectory $s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} \dots$, let $\text{time}_{\mathbf{tr}}[i]$ denote $\sum_{j=0}^i t_j$, *i.e.* the time of the i -th transition. The trace \mathbf{tr} is *time-convergent* or *zeno* if $\lim_{i \rightarrow \infty} \text{time}_{\mathbf{tr}}[i]$ is finite; otherwise it is *time-divergent* or *non-zeno*. We denote the set of time-divergent initialized traces of a timed transition system A by $\text{Timediv}(A)$, and the set of all initialized traces of A by $\text{Traces}(A)$. A TTS is well formed if from every $s_0 \in S_0$, there exists a s_0 -trace in $\text{Timediv}(A)$. We consider only well formed TTS in the sequel. The TTS $A_{\mathbf{r}}$ *refines* or *implements* the TTS $A_{\mathbf{s}}$ (the specification) if every initialized trace of $A_{\mathbf{r}}$ is also an initialized trace of $A_{\mathbf{s}}$. We first define various quantitative notions of refinement that quantify if the behavior of an implementation TTS is “close enough” to a specification TTS. We begin by defining several metrics on trace differences and refinements.

Maximum Trace Difference Distance. Given two traces $\text{tr} = r_0 \xrightarrow{t_0} r_1 \xrightarrow{t_1} r_2 \dots$ and $\text{tr}' = s_0 \xrightarrow{t'_0} s_1 \xrightarrow{t'_1} s_2 \dots$, the maximum trace difference distance $\mathcal{D}_{\text{MaxDiff}}(\text{tr}, \text{tr}')$ is defined by

$$\mathcal{D}_{\text{MaxDiff}}(\text{tr}, \text{tr}') = \begin{cases} \infty & \text{if } r_n \neq s_n \\ & \text{for some } n \\ \sup_n \{|\text{time}_{\text{tr}}[n] - \text{time}_{\text{tr}'}[n]|\} & \text{otherwise} \end{cases}$$

The distance $\mathcal{D}_{\text{MaxDiff}}(\text{tr}, \text{tr}')$ indicates the maximum time discrepancy between matching observations in the two traces tr and tr' .

Proposition 1. *The function $\mathcal{D}_{\text{MaxDiff}}()$ is a metric on timed traces.* \square

Refinement Distance Induced by $\mathcal{D}_{\text{MaxDiff}}$. The trace difference metric $\mathcal{D}_{\text{MaxDiff}}$ induces a *refinement distance* between two TTS. Given two timed transition systems A_{r} (the refined system) and A_{s} (the specification), with initial state sets $S_{\text{r}}, S_{\text{s}}$ respectively, the *refinement distance* of A_{r} with respect to A_{s} induced by $\mathcal{D}_{\text{MaxDiff}}$ is given by

$$\mathcal{R}_{\text{MaxDiff}}(A_{\text{r}}, A_{\text{s}}) = \sup_{\text{tr}_{q_{\text{r}}}} \inf_{\text{tr}_{q_{\text{s}}}} \{ \mathcal{D}_{\text{MaxDiff}}(\text{tr}_{q_{\text{r}}}, \text{tr}_{q_{\text{s}}}) \}$$

where $\text{tr}_{q_{\text{r}}}$ (respectively, $\text{tr}_{q_{\text{s}}}$) is a q_{r} -trace (respectively, q_{s} -trace) for some $q_{\text{r}} \in S_{\text{r}}$ (respectively, $q_{\text{s}} \in S_{\text{s}}$). Notice that this refinement distance is asymmetric: it is a *directed distance*[AFS09]. The refinement distance $\mathcal{R}_{\text{MaxDiff}}(A_{\text{r}}, A_{\text{s}})$ indicates quantitatively how well initialized traces in A_{s} match corresponding initialized traces in A_{r} with respect to the $\mathcal{D}_{\text{MaxDiff}}$ trace difference metric.

Proposition 2. *The function $\mathcal{R}_{\text{MaxDiff}}()$ is a directed metric on timed transition systems.* \square

We next define several other trace difference metrics, which in turn induce their own refinement distances on TTS.

Limit-Maximum Trace Difference Distance. Given two traces $\text{tr} = r_0 \xrightarrow{t_0} r_1 \xrightarrow{t_1} r_2 \dots$ and $\text{tr}' = s_0 \xrightarrow{t'_0} s_1 \xrightarrow{t'_1} s_2 \dots$, the limit-maximum trace difference distance $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}, \text{tr}')$ is defined by $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}, \text{tr}') =$

$$\begin{cases} \infty & \text{if } r_n \neq s_n \\ & \text{for some } n \\ \lim_{M \rightarrow \infty} \sup_{n \geq M} \{|\text{time}_{\text{tr}}[n] - \text{time}_{\text{tr}'}[n]|\} & \text{otherwise} \end{cases}$$

The distance $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}, \text{tr}')$ indicates the limit-maximum time discrepancy between matching observations in the two traces tr and tr' . That is, it indicates the eventual “steady state” maximum time discrepancy, ignoring any initial spikes in the time discrepancy between the two traces (we still require all observations to be matched).

In the following lemma, we view limits as being on the extended real line.

Lemma 1. *Let a_n and b_n both be non-decreasing or both be non-decreasing sequences of real numbers for $n \geq 0$. Then $\lim_{n \rightarrow \infty} (a_n)$ and $\lim_{n \rightarrow \infty} (b_n)$ both exist and*

$$\lim_{n \rightarrow \infty} (a_n) + \lim_{n \rightarrow \infty} (b_n) = \lim_{n \rightarrow \infty} (a_n + b_n) \quad \square$$

Lemma 2. Let a_n and b_n be real numbers for $n \geq 0$ and let $M \geq 0$. Then

$$\sup_{n \geq M} \{a_n\} + \sup_{n \geq M} \{b_n\} \geq \sup_{n \geq M} \{a_n + b_n\} \quad \square$$

Proposition 3. The function $\mathcal{D}_{\text{LimMaxDiff}}()$ is a metric on timed traces.

Proof. We prove $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_2) + \mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_2, \text{tr}_3) \geq \mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_3)$.

If all the observation sequences of $\text{tr}_1, \text{tr}_2, \text{tr}_3$ are not the same, or if $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_2)$ or $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_2, \text{tr}_3)$ is infinite, then the claim is straightforward. So assume that the observation sequences of the three traces are the same and that $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_2)$ and $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_2, \text{tr}_3)$ are both finite. We have $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_2) + \mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_2, \text{tr}_3)$

$$\begin{aligned} &= \lim_{M \rightarrow \infty} \sup_{n \geq M} \{|\text{time}_{\text{tr}_1}[n] - \text{time}_{\text{tr}_2}[n]|\} + \\ &\quad \lim_{M \rightarrow \infty} \sup_{n \geq M} \{|\text{time}_{\text{tr}_2}[n] - \text{time}_{\text{tr}_3}[n]|\} \\ &= \lim_{M \rightarrow \infty} \left(\sup_{n \geq M} \{|\text{time}_{\text{tr}_1}[n] - \text{time}_{\text{tr}_2}[n]|\} + \sup_{n \geq M} \{|\text{time}_{\text{tr}_2}[n] - \text{time}_{\text{tr}_3}[n]|\} \right) \\ &\quad \text{by Lemma 1.} \\ &\geq \lim_{M \rightarrow \infty} \left(\sup_{n \geq M} \left\{ (|\text{time}_{\text{tr}_1}[n] - \text{time}_{\text{tr}_2}[n]|) + (|\text{time}_{\text{tr}_2}[n] - \text{time}_{\text{tr}_3}[n]|) \right\} \right) \text{ by Lemma 2.} \\ &\geq \lim_{M \rightarrow \infty} \sup_{n \geq M} \{|\text{time}_{\text{tr}_1}[n] - \text{time}_{\text{tr}_3}[n]|\} \\ &= \mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_3) \end{aligned} \quad \square$$

Refinement Distance Induced by $\mathcal{D}_{\text{LimMaxDiff}}$. The trace difference metric $\mathcal{D}_{\text{LimMaxDiff}}$ induces the refinement distance $\mathcal{R}_{\text{LimMaxDiff}}(A_{\text{r}}, A_{\text{s}})$. Since we are interested in the long run steady state time discrepancy, we consider only time-divergent traces in A_{r} , if such traces exist. Formally, given two timed transition systems $A_{\text{r}}, A_{\text{s}}$, with initial state sets $S_{\text{r}}, S_{\text{s}}$ respectively, the refinement distance of A_{r} with respect to A_{s} induced by $\mathcal{D}_{\text{LimMaxDiff}}$ is given by

$$\mathcal{R}_{\text{LimMaxDiff}}(A_{\text{r}}, A_{\text{s}}) = \sup_{\text{tr}_{q_{\text{r}}} \in \text{Timediv}(A_{\text{r}})} \inf_{\text{tr}_{q_{\text{s}}}} \{\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_{q_{\text{r}}}, \text{tr}_{q_{\text{s}}})\}$$

where $\text{tr}_{q_{\text{r}}}$ (respectively, $\text{tr}_{q_{\text{s}}}$) is a q_{r} -trace (respectively, q_{s} -trace) for some $q_{\text{r}} \in S_{\text{r}}$ (respectively, $q_{\text{s}} \in S_{\text{s}}$). Note that we do not need to put any time-divergence requirement on the traces from A_{s} ; the “inf” operator ensures that only time-divergent traces are considered when available ($\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_{q_{\text{r}}}, \text{tr}_{q_{\text{s}}})$ is infinite if one trace is time-divergent and the other zeno). Also note that we did not place any time-divergence requirements in $\mathcal{R}_{\text{MaxDiff}}()$ as it does not have an affect on the value of the function.

Proposition 4. The function $\mathcal{R}_{\text{LimMaxDiff}}()$ is a directed metric on timed transition systems.

Proof. We prove $\mathcal{R}_{\text{LimMaxDiff}}(A_1, A_2) + \mathcal{R}_{\text{LimMaxDiff}}(A_2, A_3) \geq \mathcal{R}_{\text{LimMaxDiff}}(A_1, A_3)$.

The interesting case is when both $\mathcal{R}_{\text{LimMaxDiff}}(A_1, A_2)$ and $\mathcal{R}_{\text{LimMaxDiff}}(A_2, A_3)$ are finite. Let $\mathcal{R}_{\text{LimMaxDiff}}(A_1, A_2) = K_{1,2}$ and let $\mathcal{R}_{\text{LimMaxDiff}}(A_2, A_3) = K_{2,3}$. Consider any $\text{tr}_1 \in$

$\text{Timediv}(A_1)$. Since $K_{1,2} = \sup_{\text{tr}_{q_1} \in \text{Timediv}(A_1)} \inf_{\text{tr}_{q_2} \in \text{Traces}(A_2)} \{\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_{q_1}, \text{tr}_{q_2})\}$, we have that $K_{1,2} \geq \inf_{\text{tr}_{q_2} \in \text{Traces}(A_2)} \{\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_{q_2})\}$. Hence we have that for any given $\epsilon > 0$, there exists $\text{tr}_2 \in \text{Traces}(A_2)$ such that $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_2) < K_{1,2} + \epsilon$. Now, tr_2 must be time divergent (*i.e.* $\text{tr}_2 \in \text{Timediv}(A_2)$), otherwise $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_2)$ is not finite. Using a similar argument, we have that there exists a trace $\text{tr}_3 \in \text{Traces}(A_3)$ such that $\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_2, \text{tr}_3) < K_{2,3} + \epsilon$.

Since

$$\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_2) + \mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_2, \text{tr}_3) \geq \mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_3)$$

we have that

$$\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_3) < K_{1,2} + K_{2,3} + 2 \cdot \epsilon$$

Since this holds for any $\epsilon > 0$, we have that

$$\inf_{\text{tr}_{q_3} \in \text{Traces}(A_3)} \{\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_1, \text{tr}_{q_3})\} \leq K_{1,2} + K_{2,3}$$

And since this inequality holds for any $\text{tr}_1 \in \text{Timediv}(A_1)$, we have

$$\sup_{\text{tr}_{q_1} \in \text{Timediv}(A_1)} \inf_{\text{tr}_{q_3} \in \text{Traces}(A_3)} \{\mathcal{D}_{\text{LimMaxDiff}}(\text{tr}_{q_1}, \text{tr}_{q_3})\} \leq K_{1,2} + K_{2,3} \quad \square$$

Limit-Average Trace Difference Distance. Given two traces $\text{tr} = r_0 \xrightarrow{t_0} r_1 \xrightarrow{t_1} r_2 \dots$ and $\text{tr}' = s_0 \xrightarrow{t'_0} s_1 \xrightarrow{t'_1} s_2 \dots$, the limit-average trace difference distance $\mathcal{D}_{\text{AvgDiff}}(\text{tr}, \text{tr}')$ is defined by $\mathcal{D}_{\text{AvgDiff}}(\text{tr}, \text{tr}') =$

$$\begin{cases} \infty & \text{if } r_j \neq s_j \text{ for some } j \\ \lim_{M \rightarrow \infty} \left(\sup_{n \geq M} \left\{ \frac{\sum_{i=0}^n (|\text{time}_{\text{tr}}[i] - \text{time}_{\text{tr}'}[i]|)}{n} \right\} \right) & \text{otherwise} \end{cases}$$

The distance $\mathcal{D}_{\text{AvgDiff}}(\text{tr}, \text{tr}')$ indicates the long run average of the time discrepancies between the two traces.

Proposition 5. *The function $\mathcal{D}_{\text{AvgDiff}}()$ is a metric on timed traces.*

Proof. We prove $\mathcal{D}_{\text{AvgDiff}}(\text{tr}_1, \text{tr}_2) + \mathcal{D}_{\text{AvgDiff}}(\text{tr}_2, \text{tr}_3) \geq \mathcal{D}_{\text{AvgDiff}}(\text{tr}_1, \text{tr}_3)$.

If all the observation sequences of $\text{tr}_1, \text{tr}_2, \text{tr}_3$ are not the same, or if $\mathcal{D}_{\text{AvgDiff}}(\text{tr}_1, \text{tr}_2)$ or $\mathcal{D}_{\text{AvgDiff}}(\text{tr}_2, \text{tr}_3)$ is infinite, then the claim is straightforward. So assume that the observation sequences of the three traces are the same and that $\mathcal{D}_{\text{AvgDiff}}(\text{tr}_1, \text{tr}_2)$ and $\mathcal{D}_{\text{AvgDiff}}(\text{tr}_2, \text{tr}_3)$ are both

finite. We have $\mathcal{D}_{\text{AvgDiff}}(\text{tr}_1, \text{tr}_2) + \mathcal{D}_{\text{AvgDiff}}(\text{tr}_2, \text{tr}_3)$

$$\begin{aligned}
&= \lim_{M \rightarrow \infty} \left(\sup_{n \geq M} \left\{ \frac{\sum_{i=0}^n (|\text{time}_{\text{tr}_1}[i] - \text{time}_{\text{tr}_2}[i]|)}{n} \right\} \right) + \\
&\quad \lim_{M \rightarrow \infty} \left(\sup_{n \geq M} \left\{ \frac{\sum_{i=0}^n (|\text{time}_{\text{tr}_2}[i] - \text{time}_{\text{tr}_3}[i]|)}{n} \right\} \right) \\
&= \lim_{M \rightarrow \infty} \left(\sup_{n \geq M} \left\{ \frac{\sum_{i=0}^n (|\text{time}_{\text{tr}_1}[i] - \text{time}_{\text{tr}_2}[i]|)}{n} \right\} + \sup_{n \geq M} \left\{ \frac{\sum_{i=0}^n (|\text{time}_{\text{tr}_2}[i] - \text{time}_{\text{tr}_3}[i]|)}{n} \right\} \right) \\
&\quad \text{by Lemma 1.} \\
&\geq \lim_{M \rightarrow \infty} \left(\sup_{n \geq M} \left\{ \frac{\sum_{i=0}^n (|\text{time}_{\text{tr}_1}[i] - \text{time}_{\text{tr}_2}[i]|)}{\sum_{i=0}^n (|\text{time}_{\text{tr}_2}[i] - \text{time}_{\text{tr}_3}[i]|)} + \right\} \right) \text{ by Lemma 2.} \\
&\geq \lim_{M \rightarrow \infty} \left(\sup_{n \geq M} \left\{ \frac{\sum_{i=0}^n (|\text{time}_{\text{tr}_1}[i] - \text{time}_{\text{tr}_3}[i]|)}{n} \right\} \right) \\
&= \mathcal{D}_{\text{AvgDiff}}(\text{tr}_1, \text{tr}_3) \quad \square
\end{aligned}$$

Refinement Distance Induced by $\mathcal{D}_{\text{AvgDiff}}$. The trace difference metric $\mathcal{D}_{\text{AvgDiff}}$ induces the refinement distance $\mathcal{R}_{\text{AvgDiff}}(A_{\text{r}}, A_{\text{s}})$. As in the definition of $\mathcal{R}_{\text{add}}(A_{\text{r}}, A_{\text{s}})$, we only consider time-divergent traces from A_{r} when available. Formally, given two timed transition systems $A_{\text{r}}, A_{\text{s}}$, with initial state sets $S_{\text{r}}, S_{\text{s}}$ respectively, the refinement distance of A_{r} with respect to A_{s} induced by $\mathcal{D}_{\text{AvgDiff}}$ is given by

$$\mathcal{R}_{\text{AvgDiff}}(A_{\text{r}}, A_{\text{s}}) = \sup_{\text{tr}_{q_{\text{r}}} \in \text{Timediv}(A_{\text{r}})} \inf_{\text{tr}_{q_{\text{s}}}} \{ \mathcal{D}_{\text{AvgDiff}}(\text{tr}_{q_{\text{r}}}, \text{tr}_{q_{\text{s}}}) \}$$

where $\text{tr}_{q_{\text{r}}}$ (respectively, $\text{tr}_{q_{\text{s}}}$) is a q_{r} -trace (respectively, q_{s} -trace) for some $q_{\text{r}} \in S_{\text{r}}$ (respectively, $q_{\text{s}} \in S_{\text{s}}$).

Proposition 6. *The function $\mathcal{R}_{\text{AvgDiff}}()$ is a directed metric on timed transition systems.*

Proof. The proof is similar to Proposition 4. □

A Note on Zeno-Asymmetry in Refinement Metrics. There appears to be an asymmetry in the definitions for refinement metrics with respect to zenoness as only zeno behaviors of A_{r} are given special treatment. This is because in case of zeno behavior by the specification, our definitions automatically give a value of ∞ , which is the correct notion. That is, for $\Psi \in \{\mathcal{D}_{\text{MaxDiff}}, \mathcal{D}_{\text{LimMaxDiff}}, \mathcal{D}_{\text{AvgDiff}}\}$, we have $\Psi(\text{tr}_{q_{\text{r}}}, \text{tr}_{q_{\text{s}}}) = \infty$ if $\text{tr}_{q_{\text{r}}}$ is time divergent, and $\text{tr}_{q_{\text{s}}}$ is time convergent.

3 Timed Simulation Relations

The general trace inclusion problem for timed systems is undecidable [AD94], simulation relations allow us to restrict our attention to a computable relation.

Timed Simulation Relations. Let A_{r} and \mathcal{T}_{s} be timed transition systems. A binary relation $\preceq \subseteq S_{\text{r}} \times S_{\text{s}}$ is a *timed simulation* if $s_{\text{r}} \preceq s_{\text{s}}$ implies the following conditions:

1. $\mu(s_\tau) = \mu(s_s)$.
2. If $s_\tau \xrightarrow{t} s'_\tau$, then there exists s'_s such that $s_s \xrightarrow{t} s'_s$, and $s'_\tau \preceq s'_s$.

The state s_τ is timed simulated by the state s_s if there exists a timed simulation \preceq such that $s_\tau \preceq s_s$. A binary relation \equiv is a *timed bisimulation* if it is a symmetric timed simulation. Two states s_τ and s_s are timed bisimilar if there exists a timed bisimulation \equiv with $s_\tau \equiv s_s$. Timed bisimulation is stronger than timed simulation which in turn is stronger than trace inclusion. If state s_τ is timed simulated by state s_s , then every s_τ -trace is also a s_s -trace.

Untimed Simulation Relations. *Untimed* simulation and bisimulation relations are defined analogously to timed simulation and bisimulation relations by ignoring the duration of time steps. Formally, a binary relation $\preceq_u \subseteq S_\tau \times S_s$ is an (untimed) simulation if $s_\tau \preceq_u s_s$ implies the following conditions:

1. $\mu(s_\tau) = \mu(s_s)$.
2. If $s_\tau \xrightarrow{t} s'_\tau$, then there exists s'_s and $t' \in \mathbb{R}^+$ such that $s_s \xrightarrow{t'} s'_s$, and $s'_\tau \preceq s'_s$.

A symmetric untimed simulation relation is called an untimed bisimulation.

Timed simulation and bisimulation require that times be matched exactly. This is often too strict a requirement, especially since timed models are approximations of the real world. On the other hand, untimed simulation and bisimulation relations ignore the times on moves altogether. Analogous to the notions of quantitative refinement presented in Section 2, we will define quantitative notions of simulation functions which lie in between these extremes in Section 5. We will define quantitative simulation functions in a game theoretic framework. The motivation for the game theoretic framework for simulation relations is presented next.

Timed and Untimed Simulation Games. We present an alternative equivalent game theoretic view of timed simulation (a similar view exists for untimed simulation). Given two timed transition systems A_τ and A_s , consider a two player turn-based bipartite timed transition game structure $\mathfrak{G}_t(A_\tau, A_s)$ with state space $(S_\tau \times S_s \times \{1\}) \cup (S_\tau \times S_s \times \{2\})$ (the full formal definitions of game structures will be presented in Section 4). The states of player 1 (the antagonist) are $S_\tau \times S_s \times \{2\}$ and player-2 (the protagonist) states are $S_\tau \times S_s \times \{1\}$. The transitions are:

Player-2 transitions. $\langle s_\tau, s_s, 2 \rangle \xrightarrow{\Delta_\tau} \langle s'_\tau, s_s, 1 \rangle$ such that $s_\tau \xrightarrow{\Delta_\tau} s'_\tau$ is a valid transition in A_τ .

Player-1 transitions. $\langle s_\tau, s_s, 1 \rangle \xrightarrow{\Delta_s} \langle s_\tau, s'_s, 2 \rangle$ such that $s_s \xrightarrow{\Delta_s} s'_s$ is a valid transition in A_s .

To decide if s_s time-simulates s_τ , we play the following game. Let $\langle s_\tau, s_s, 2 \rangle$ be the initial state such that $\mu(s_\tau) = \mu(s_s)$. Player-2 picks a transition of some duration Δ_τ from this state and moves to some state $\langle s'_\tau, s_s, 1 \rangle$. From $\langle s'_\tau, s_s, 1 \rangle$, player 1 then picks a transition of duration Δ_s such that $\Delta_s = \Delta_\tau$ and moves to $\langle s_\tau, s'_s, 2 \rangle$ such that $\mu(s'_s) = \mu(s_s)$. If no such transition exists, then player 1 loses. If the game can proceed forever without player-1 losing, then player 2 loses and player 1 wins. If player 1 wins starting from $\langle s_\tau, s_s, 2 \rangle$, then s_s time-simulates s_τ . For untimed simulation, we ignore the time durations of the moves (and player 1 can pick transitions of any duration from A_s). We denote the two player turn-based bipartite *untimed* transition game as $\mathfrak{G}_u(A_\tau, A_s)$.

4 Finite-state Game Graphs

We will define the values of *quantitative timed simulation functions* in Section 5 through game theoretic formulations of problems for finite-state game graphs. In this section, we first present the basic background on finite-state game graphs, and the relevant known results; then introduce new game theoretic objectives (that were not studied before but are required for quantitative timed simulation functions) and present solutions for the new objectives.

4.1 Basic Definitions and Known Results

In this section we present definitions of finite game graphs, plays, strategies, objectives, notion of winning and the decision problems.

Game graphs. A *game graph* $G = \langle Q, E \rangle$ consists of a finite set Q of states partitioned into player-1 states Q_1 and player-2 states Q_2 (i.e., $Q = Q_1 \cup Q_2$ and $Q_1 \cap Q_2 = \emptyset$), and a set $E \subseteq Q \times Q$ of edges such that for all $q \in Q$, there exists (at least one) $q' \in Q$ such that $(q, q') \in E$. A *player-1 game* is a game graph where $Q_1 = Q$ and $Q_2 = \emptyset$. The subgraph of G induced by $S \subseteq Q$ is the graph $\langle S, E \cap (S \times S) \rangle$ (which is not a game graph in general); the subgraph induced by S is a game graph if for all $s \in S$ there exist $s' \in S$ such that $(s, s') \in E$.

Plays and strategies. A game on G starting from a state $q_0 \in Q$ is played in rounds as follows. If the game is in a player-1 state, then player 1 chooses the successor state from the set of outgoing edges; otherwise the game is in a player-2 state, and player 2 chooses the successor state from the set of outgoing edges. The game results in a *play* from q_0 , i.e., an infinite path $\rho = q_0 q_1 \dots$ such that $(q_i, q_{i+1}) \in E$ for all $i \geq 0$. The prefix of length n of ρ is denoted by $\rho(n) = q_0 \dots q_n$. A *strategy* for player 1 is a function $\pi_1 : Q^* Q_1 \rightarrow Q$ such that $(q, \pi_1(\rho \cdot q)) \in E$ for all $\rho \in Q^*$ and $q \in Q_1$. An *outcome* of π_1 from q_0 is a play $q_0 q_1 \dots$ such that $\pi_1(q_0 \dots q_i) = q_{i+1}$ for all $i \geq 0$ such that $q_i \in Q_1$. Strategy and outcome for player 2 are defined analogously. A player-1 strategy is *memoryless* if it is independent of the history and depends only on the current state, and hence can be described as a function $\pi_1 : Q_1 \rightarrow Q$. Memoryless strategies for player 2 are defined analogously. We denote by Π_1 and Π_2 the set of strategies for player 1 and player 2, respectively. Given a starting state q , a strategy π_1 for player 1 and a strategy π_2 for player 2, we have a unique play $q_0 q_1 q_2 \dots$, such that $q_0 = q$ and for all $i \geq 0$ (i) if q_i is a player 1 state, then $q_{i+1} = \pi_1(q_0, q_1, \dots, q_i)$; and (ii) if q_i is a player 2 state, then $q_{i+1} = \pi_2(q_0, q_1, \dots, q_i)$. We denote the unique play as $\rho(\pi_1, \pi_2, q)$.

Objectives. In this work we will consider both qualitative and quantitative objectives. We first introduce qualitative objectives that we will use in our work. A *qualitative objective* for G is a set $\phi \subseteq Q^\omega$ of winning plays. For a play ρ , we denote by $\text{Inf}(\rho)$ the set of states that occur infinitely often in ρ . We consider Büchi objectives, and its dual coBüchi objectives which are defined as follows. A Büchi objective consists of a set B of Büchi states, and requires that the set B is visited infinitely often. Formally, the Büchi objective defines the following set of winning plays: $\text{Büchi}(B) = \{\rho \mid \text{Inf}(\rho) \cap B \neq \emptyset\}$. Dually the coBüchi objective consists of a set C of coBüchi states and requires that states outside C be visited only finitely often, and defines the set $\text{coBüchi}(C) = \{\rho \mid \text{Inf}(\rho) \subseteq C\}$ of winning plays. When we will consider qualitative objectives, the objective of player 1 will be disjunction of two coBüchi objectives, and the objective of player 2 will be the complement (conjunction of two Büchi objectives). We now introduce several quantitative objectives.

Quantitative objectives. A *quantitative objective* for G is a function $f : Q^\omega \rightarrow \mathbb{R}$ that maps every play to a real-valued number (in contrast a qualitative objective can be interpreted as a function $\phi : Q^\omega \rightarrow \{0, 1\}$ that maps plays to Boolean rewards, with 1 for winning plays). Let $w : E \rightarrow \mathbb{Z}$ be a *weight function* and let us denote by W the largest weight (in absolute value) according to w . For a prefix $\rho(n) = q_0 q_1 \dots q_n$ of a play we denote by $\text{Sum}(w)(\rho(n)) = \sum_{i=0}^{n-1} w(q_i, q_{i+1})$ the sum of the weights of the prefix. The *debit-sum* level at the end of the prefix $\rho(n)$ is defined by

$$\text{DebSum}(w)(\rho(n)) = \max(0, -\sum_{i=0}^{n-1} w(q_i, q_{i+1}))$$

Note the negative sign in the definition. The debit-sum level denotes the amount by which the accumulated sum of the weights has dipped below 0 at the end of $\rho(n)$ (if the sum of the weights is positive, *i.e.* there is a credit, then the debit-sum level is defined to be 0). We will consider the following objective functions.

Debit-sum level. For a play ρ , the debit-sum level is the maximal debit-sum level that occurs in it. Formally, for a play ρ and the weight function w we have $\text{DebSum}(w)(\rho) = \sup_n \text{DebSum}(w)(\rho(n)) = \inf\{v_0 \mid \forall n \geq 0. v_0 + \text{Sum}(w)(\rho(n)) \geq 0\}$.

Eventual debit-sum level. For a play ρ , the eventual debit-sum level is the maximal debit-sum level that occurs after some point on in the play. Formally, for a play ρ and the weight function w we have $\text{EvDebSum}(w)(\rho) = \limsup_{n \rightarrow \infty} \text{DebSum}(w)(\rho(n)) = \lim_{M \rightarrow \infty} \sup_{n \geq M} \text{DebSum}(w)(\rho(n)) = \inf\{v_0 \mid \exists n_0 \geq 0. \forall n \geq n_0. v_0 + \text{Sum}(w)(\rho(n)) \geq 0\}$.

Average weight. The mean-payoff (or limit-average weight) objective function on a play $\rho = q_0 q_1 \dots$ is the long-run average of the weights of the play, *i.e.*, $\text{Avg}(w)(\rho) = \limsup_{n \rightarrow \infty} \frac{1}{n} \cdot \text{Sum}(w)(\rho(n))$.

Average debit-sum. Along with the previous objective, we introduce a new objective function, which we call the average debit-sum level that assigns to every play the long-run average of the debit-sum levels. Formally, $\text{AvDebSum}(w)(\rho) = \limsup_{n \rightarrow \infty} \frac{\sum_{i=0}^n \text{DebSum}(w)(\rho(i))}{n}$. Note that since the debit-sum level is defined to be 0 if the accumulated sum is positive (*i.e.* a positive credit-sum level), a positive credit-sum cannot cancel out a positive debit-sum in the averaging process in $\text{AvDebSum}(w)(\rho)$. Observe that in contrast to mean-payoff objective that is the average of the weights, the average debit-sum has the flavor of the average of the partial sums of the weights.

In the sequel, when the weight function w is clear from context we will omit it and simply write $\text{Sum}(\rho(n))$ and $\text{Avg}(\rho)$, and so on. For each of the above quantitative objective, we will consider a version of the quantitative objective that is a disjunction with a coBüchi objective. Formally for a quantitative objective f and coBüchi objective $\text{coBüchi}(C)$, the quantitative objective that is the disjunction of the two objectives is defined as follows for a play ρ : if $\rho \in \text{coBüchi}(C)$, then the objective function assigns value 0 to ρ , otherwise it assigns value $f(\rho)$. We will refer to the corresponding version of the quantitative objectives with disjunction with coBüchi objective as DebSumCB , EvDebSumCB , AvgCB , and AvDebSumCB , respectively (and when the weight function w and the coBüchi set C is clear from the context we drop them for simplicity).

Winning strategies, optimal value and optimal strategies. A player-1 strategy π_1 is *winning* (we also say that player 1 is winning, or that q is a winning state) in a state q for a qualitative objective ϕ if $\rho \in \phi$ for all outcomes ρ of π_1 from q . The optimal value for a quantitative objective is the minimal value that player 1 can guarantee against all strategies of player 2. Formally, for a quantitative f that maps plays to real-valued reward, the optimal value $\text{Opt}(f)(q)$ at state q is defined as

$$\text{Opt}(f)(q) = \inf_{\pi_1 \in \Pi_1} \sup_{\pi_2 \in \Pi_2} f(\rho(\pi_1, \pi_2, q)).$$

A strategy for player 1 is optimal if it achieves the optimal value against all strategies of player 2, *i.e.*, a strategy π_1^* is optimal if we have

$$\text{Opt}(f)(q) = \sup_{\pi_2 \in \Pi_2} f(\rho(\pi_1^*, \pi_2, q)).$$

We now present a theorem that summarizes known results about Büchi and coBüchi games, debit sum (minimal initial credit for energy games), and mean-payoff games. The results of Büchi

and coBüchi objectives follow from [EJ91], the results for debit sum games credit follows from the results on energy games of [CD10], and the result for mean-payoff games follows from [Bou+11; CHJ05] (also note that in [Bou+11; CD10; CHJ05] player 1 has conjunction of energy (or mean-payoff) with parity objectives, whereas in our setting player 1 has the disjunction of energy (or mean-payoff) with parity, and thus the roles of player 1 and player 2 in this work is exchanged as compared to [Bou+11; CD10; CHJ05]).

Theorem 1. *The following assertions hold for finite-state game graphs.*

1. *The set of winning states in games with disjunction of two coBüchi objectives can be computed in time $O(|Q| \cdot |E|)$, and memoryless winning strategies exist for player 1 and winning strategies of player 2 require one-bit memory (from their respective winning states).*
2. *The optimal value for debit-sum function with coBüchi disjunction can be computed in time $O(|Q|^2 \cdot |E| \cdot W)$, and memoryless optimal strategies exist for player 1 and optimal strategies for player 2 require finite memory. If the optimal value is finite, then the optimal value is at most $|Q| \cdot |W|$.*
3. *The optimal value for limit-average function with coBüchi disjunction can be computed in time $O(|Q|^2 \cdot |E| \cdot W)$, and memoryless optimal strategies exist for player 1 and the optimal strategies of player 2 may require infinite memory.* \square

4.2 New Results and Algorithms

In this section we will present two solutions for problems on finite-state game graphs. The first solution is for games with minimal initial credit for eventual survival, and the second solution for average-sum objectives.

Eventual Debit-Sum Level Objectives We will solve the problem by a reduction to a coBüchi game. We start with a lemma that is required for the reduction.

Lemma 3. *For all game graphs with a weight function w , the following assertions hold:*

1. *The optimal value of the eventual debit sum level is at most the optimal value of the debit sum level objective i.e., for all states q we have*

$$\text{Opt}(\text{EvDebSum})(q) \leq \text{Opt}(\text{DebSum})(q);$$

2. *If the optimal value of the debit sum level objective is infinite, then the optimal value of the eventual debit sum level is also infinite.*

Proof. The first item follows from definition. The proof of the second item is as follows: if we have a sequence $(x_n)_{n \geq 0}$ of integers, then $\sup x_n = \infty$ iff $\limsup x_n = \infty$. Considering $(x_n)_{n \geq 0}$ to be the sequence $(\text{Sum}(\rho(n)))_{n \geq 0}$ we obtain the result for all plays, and hence the result follows. \square

Reduction to coBüchi games. The solution for the optimal value is obtained as follows: (1) We compute $\text{Opt}(\text{DebSum})(q)$ using algorithms of Theorem 1, and if $\text{Opt}(\text{DebSum})(q)$ is infinite, then $\text{Opt}(\text{EvDebSum})(q)$ is infinite (by Lemma 3); (2) if $\text{Opt}(\text{DebSum})(q)$ is finite, then by Lemma 3 we have $\text{Opt}(\text{EvDebSum})(q)$ is finite and by Theorem 1 we have $\text{Opt}(\text{EvDebSum})(q) \leq |Q| \cdot W$. If $\text{Opt}(\text{EvDebSum})(q)$ is finite, for $0 \leq D \leq |Q| \cdot W$ the procedure to check whether

$\text{Opt}(\text{EvDebSum})(q) \leq D$ is as follows: we construct a coBüchi game where we keep track of the current sum of weights; and since the optimum value for the debit sum level objective is at most $|Q| \cdot W$, then player 1 can ensure that the sum of the weights never decreases below $-|Q| \cdot W$. Moreover, any optimal strategy for player 1 must ensure that a state where the optimal value is ∞ is never reached. If the sum of the weights exceeds $|Q| \cdot W$, then a optimal strategy for the debit sum level objective ensures that the sum never falls below 0 afterwards. Hence we only need to keep track of the sum of weights that lie between $-|Q| \cdot W$ and $|Q| \cdot W$. If the sum of the weights is above $-D$, then we call the state a coBüchi state, otherwise it is a bad state for the coBüchi objective. The goal of player 1 is the coBüchi objective, which is equivalently the objective to ensure that from some point on the sum of the weights is always above $-D$. Using a binary search for D for values between 0 and $|Q| \cdot W$ we obtain the optimal value. Also observe that the games we construct for the binary searches have at most $O(|Q|^2 \cdot W)$ states and $O(|E| \cdot |Q| \cdot W)$ edges. Note that for disjunction with coBüchi objective, we have the same reduction as above, but in the end we obtain a game with disjunction of two coBüchi objectives.

Theorem 2. *The optimal player-1 strategy, and the optimal value $\text{Opt}(\text{EvDebSumCB})(q)$ for the eventual debit sum level objective with coBüchi disjunction can be computed in time $O(|Q|^3 \cdot |E| \cdot W^2 \cdot \log(|Q| \cdot W))$.* \square

The next example illustrates the difference between debit sum level and eventual debit sum level objectives.

Example 7 (Debit sum vs eventual debit sum level). Consider the game graph G_0 in Figure 6. The

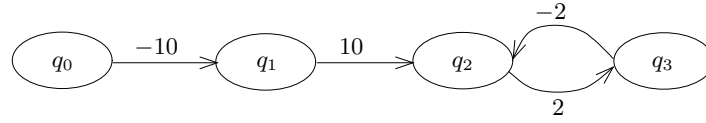


Fig. 6. Game Graph G_0

game G_0 has only one play from q_0 , namely, $q_0 \rightarrow q_1 \rightarrow (q_2 \rightarrow q_3 \rightarrow)^\omega$. It can be seen that $\text{Opt}(\text{DebSum})(q_0)$ is 10 as a debit level of 10 is seen on the transition from q_0 to q_1 . However, $\text{Opt}(\text{EvDebSum})(q_0)$ is only 2, as the debit level 10 occurs only once in the play. The debit level 2 however occurs infinitely often in the play. Thus, $\text{Opt}(\text{EvDebSum})(q_0)$ is 2. \square

Average Debit-Sum Level Objectives We start with an example that illustrates average debit-sum level objectives.

Example 8. Consider the game graph G_1 in Figure 7. The game G_1 has only one play from q_0 , namely, $(q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow)^\omega$ (and similarly only one play from any state). For this play we compute the debit sum and credit sum levels: let $\langle q, d, c \rangle$ denote the state q , and d, c the debit and credit sum levels at that point in the play (note that only either debit sum, or credit sum level can be non-zero, by definition). The play together with debit and credit sum levels is:

$$\langle q_0, 0, 0 \rangle \rightarrow (\langle q_1, 1, 0 \rangle \rightarrow \langle q_2, 0, 1 \rangle \rightarrow \langle q_0, 0, 0 \rangle \rightarrow)^\omega$$

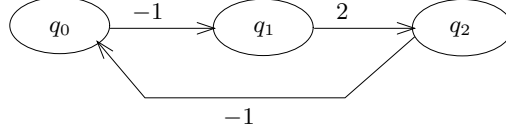


Fig. 7. Game Graph G_1

Thus the average debit sum level $\text{AvDebSum}(w)(q_0) = 1/3$. Now consider the only play from q_2 . The play annotated with debit and credit sum levels is:

$$\langle q_2, 0, 0 \rangle \rightarrow (\langle q_0, 1, 0 \rangle \rightarrow \langle q_1, 2, 0 \rangle \rightarrow \langle q_2, 0, 0 \rangle \rightarrow)^\omega$$

Note that credit levels never rise above 0 in this play. The average debit sum level $\text{AvDebSum}(w)(q_2)$ for this play is 1. Thus, where we “enter” in a cycle affects the value of the average debit sum level. \square

The next lemma is a technical lemma on integer sequences.

Lemma 4. *Let x_0, x_1, \dots be a sequence of integers. The following assertions hold.*

1. *If x_i is positive for every i , and there exist $i_0 \geq 0$ and $N > 0$ such that for all $i \geq i_0$, there exists $1 \leq m_i \leq N$ such that $x_{i+m_i} > x_i$. Then, $\lim_{M \rightarrow \infty} \left(\sup_{k > M} \left\{ \frac{\sum_{i=0}^{k-1} x_i}{k} \right\} \right) = \infty$.*
2. *Suppose (i) there exists $W < \infty$ such that for all $i \geq 0$, we have $|x_{i+1} - x_i| \leq W$; and (ii) there exist $i_0 \geq 0$ and $N > 0$ such that for all $i \geq i_0$, there exists $1 \leq m_i \leq N$ such that $x_{i+m_i} < x_i$. Then, there exists $M \geq 0$ such that $x_i \leq 0$ for all $i \geq M$.*

Proof. We present both items of the proof.

1. Consider $\sum_{i=i_0}^{i_0+\alpha \cdot N+j} x_i$ for $\alpha \geq 0$ and $0 \leq j < N$. Consider the set

$$X_\alpha = \{x_j \mid i_0 + \alpha \cdot N \leq j < i_0 + \alpha \cdot (N+1)\}$$

It can be shown by induction that for every $\alpha \geq 0$, we have: (i) there exists $x_i \in X_\alpha$, such that $x_i \geq \alpha$ (informally, the claims hold because there is an increment of at least one, starting from x_{i_0} , in every N steps); and hence, (ii) $\sum_{i=i_0}^{i_0+\alpha \cdot N+j} x_i \geq 0 + 1 + \dots + \alpha$ (since we can pick $x_i \in X_\alpha$ such that $x_i \geq \alpha$). Thus,

$$\frac{\sum_{i=i_0}^{i_0+\alpha \cdot N+j} x_i}{i_0 + \alpha \cdot N + j} \geq \frac{\alpha \cdot (\alpha + 1)}{2 \cdot (i_0 + \alpha \cdot N + j)} \geq \frac{\alpha \cdot (\alpha + 1)}{2 \cdot (i_0 + (\alpha + 1) \cdot N)}$$

for every $\alpha \geq 0$ and $0 \leq j < N$. Thus,

$$\frac{\sum_{i=i_0}^{i_0+\alpha \cdot N+j} x_i}{i_0 + \alpha \cdot N + j} \geq \frac{\alpha}{2 \cdot (\frac{i_0}{\alpha+1} + N)}$$

Therefore, for every $\alpha \geq 0$, we have

$$\sup_{k > (i_0 + \alpha \cdot N)} \left\{ \frac{\sum_{i=0}^{k-1} x_i}{k} \right\} \geq \frac{\sum_{i=0}^{i_0+\alpha \cdot N} x_i}{i_0 + \alpha \cdot N} \geq \frac{\alpha}{2 \cdot (\frac{i_0}{\alpha+1} + N)}$$

Letting $\alpha \rightarrow \infty$, we have the desired result.

2. By induction, it can be shown that for every $\alpha \geq 0$, there exists $x_{i_\alpha} \in \{x_j \mid i_o + \alpha \cdot N \leq j < i_o + \alpha \cdot (N + 1)\}$, such that $x_{i_\alpha} + \alpha \leq x_{i_0}$ (that is, x_{i_α} is at least α less than x_{i_0}). Informally, the claims hold because there is a decrement of at least one, starting from x_{i_0} , in every N steps. Consider any $\alpha > 1 + N \cdot W + x_{i_0}$. Consider the set

$$X_\alpha = \{x_j \mid i_o + \alpha \cdot N \leq j < i_o + \alpha \cdot (N + 1)\}$$

Since, $|x_{i+1} - x_i| \leq W$ for i in the given sequence, for any $x, x' \in X_\alpha$, we must have $|x - x'| \leq N \cdot W$. Also, there exists $x_\alpha \in X_\alpha$ such that $x_{i_\alpha} + \alpha \leq x_{i_0}$. Thus, for all $x \in X_\alpha$, we have

$$x + \alpha \leq x_{i_0} + N \cdot W$$

Since $\alpha > 1 + N \cdot W + x_{i_0}$, we have,

$$x + 1 + N \cdot W + x_{i_0} \leq x_{i_0} + N \cdot W$$

Rearranging, we get $x \leq -1$. Thus, for all $i > (2 + N \cdot W + x_{i_0}) \cdot N$, we have $x_i \leq -1$. \square

Corollary 1. *Consider a play $\rho = q_0 q_1 \dots$ of a finite-state game graph G . The following assertions hold.*

1. *Suppose there exist $i_0 \geq 0$ and $N > 0$ such that for all $i \geq i_0$, there exists $1 \leq m_i \leq N$ such that $\text{Sum}(\rho(i)) > \text{Sum}(\rho(i + m_i))$. Then, $\text{AvDebSum}(\rho) = \infty$.*
2. *Suppose there exist $i_0 \geq 0$ and $N > 0$ such that for all $i \geq i_0$, there exists $1 \leq m_i \leq N$ such that $\text{Sum}(\rho(i)) < \text{Sum}(\rho(i + m_i))$. Then, $\text{AvDebSum}(\rho) = 0$.*

Proof. For the first assertion, it can be shown that there exists $i'_0 \geq 0$ and $N > 0$ such that for all $i \geq i'_0$, there exists $1 \leq m_i \leq N$ such that $\text{DebSum}(\rho(i + m_i)) > \text{DebSum}(\rho(i))$. The proof of the first assertion follows from the first part of Lemma 4, and by the definition of $\text{DebSum}(\rho(n))$.

The proof of the second assertion follows from the second part of Lemma 4, and noting that if $-\text{Sum}(\rho(n)) < 0$ then $\text{DebSum}(\rho(n)) = 0$. \square

Lemma 5. *The following assertions hold: consider a weight function w , and coBüchi objective $\text{coBüchi}(C)$, and then we have*

1. *If $\text{Opt}(\text{DebSumCB})(q) = \infty$, then $\text{Opt}(\text{AvDebSumCB})(q) = \infty$.*
2. *If $\text{Opt}(\text{AvgCB})(q) > 0$, then $\text{Opt}(\text{AvDebSumCB})(q) = 0$.*

Proof. We present proof of both the items.

1. If $\text{Opt}(\text{DebSum})(q) = \infty$, then consider a finite-memory optimal strategy π_2^* for player 2 (such a strategy exist by Theorem 1). Once the strategy π_2^* is fixed we obtain a graph where only player 1 makes choices. Since $\text{Opt}(\text{DebSumCB})(q) = \infty$, it follows that for every cycle U in the graph the sum of the weights in U is negative, and there is at least one state in U that is not a coBüchi state (i.e., $U \cap (Q \setminus C) \neq \emptyset$). Since all cycles are negative the first condition of Corollary 1 is satisfied for all paths with N as the size of the graph. Moreover the coBüchi objective is also falsified. This concludes the proof of the first item.
2. The condition $\text{Opt}(\text{AvgCB})(q) > 0$ is equivalently saying that player 1 can enforce a cycle U such that the sum of weights of U is positive or $U \subseteq C$. Consider a memoryless optimal strategy for player 1 for the limit-average objective with coBüchi disjunction (such a strategy exist by

Theorem 1). Since $\text{Opt}(\text{AvgCB})(q) > 0$, it follows that in the graph obtained by fixing the strategy, for every cycle U either the sum of the weights is positive or $U \subseteq C$. Either the coBüchi objective is satisfied or if the cycle has positive weights then the second condition of Corollary 1 is satisfied. In either case the desired result of the second item follows. \square

Reduction to mean-payoff coBüchi games. We now use the above lemma to solve the average debit sum problem. Using the above lemma, and solution for $\text{Opt}(\text{DebSumCB})$ and $\text{Opt}(\text{AvgCB})$ we can identify whether $\text{Opt}(\text{AvDebSum})$ is infinite or 0. If $\text{Opt}(\text{DebSumCB})(q)$ is finite, and $\text{Opt}(\text{AvgCB})(q) = 0$, it follows that both players can play strategies to ensure that all cycles formed after their chosen strategy is fixed has sum of weights exactly equal to 0, and has a non coBüchi state. Observe that a positive cycle is only favorable for player 1 for the average debit sum objective, and a negative cycle is favorable for player 2. Hence there exist optimal plays for the average debit sum objective where for all cycles formed along the play the sum of the weights of the cycle will exactly be 0. Thus we reduce the average debit sum problem to solving a larger mean-payoff game as follows: we keep track of the current sum of weights, and since all cycles formed will have exactly 0 sums, we only need to keep track of weights from $-|Q| \cdot W$ to $|Q| \cdot W$. For the limit-average game, we construct a weight function according to the current sum of weights, i.e., if the current sum of weights is ℓ , then the weight function assigns value $\max(-\ell, 0)$. The optimal value of the constructed game with limit-average objective is the optimal value for the average debit sum objective in the original game. The constructed game has $O(|Q|^2 \cdot W)$ states, $O(|E| \cdot |Q| \cdot W)$ edges, and the maximal absolute value of the weight is $O(|Q| \cdot W)$. Thus our reduction and Theorem 1 yield the following result for average debit sum objectives.

Theorem 3. *The optimal player-1 strategy, and the optimal value $\text{Opt}(\text{AvDebSum})(q)$ for average debit sum objective with coBüchi disjunction can be computed in time $O(|Q|^6 \cdot |E| \cdot W^4)$.* \square

From debit-sum to difference-sum. An easy extension of the debit-sum objectives is instead of the sum of the weights, we consider the absolute values of the sum of the weights. We call the corresponding version as Diffsum instead of DebSum. This can be modeled as two weight functions (the original weight function and its negation), and then apply results for two-dimensional energy and mean-payoff games with disjunction with coBüchi objectives. Applying our techniques to solve eventual debit sum, and average debit sum, along with the results of [Cha+10; Cha10; VR11] we obtain the following result.

Theorem 4. *The optimal player-1 strategy, and the optimal value for difference-sum function with coBüchi disjunction, $\text{Opt}(\text{DiffSumCB})(q)$, the optimal value $\text{Opt}(\text{EvDiffSumCB})(q)$ for the eventual difference sum level objective with coBüchi disjunction, and the optimal value $\text{Opt}(\text{AvDiffSumCB})(q)$ for average difference sum objective with coBüchi disjunction, can all be computed in $O(\text{poly}(Q, E, W))$ time, where poly is a polynomial function.* \square

5 Quantitative Timed Simulation Functions

In this section, we first define quantitative timed simulation functions for timed transition systems in Subsection 5.1 in a game theoretic framework. We next present the model of timed automata in Subsection 5.2. finally, we present algorithms for computing the quantitative simulation functions in Subsection 5.3.

5.1 Quantitative Timed Simulation Functions from Timed Games

Timed Transition Game Structures. A *timed transition game structure* is a tuple $\mathfrak{S}_t = \langle S, \rightarrow \rangle$ where

- S is the set of states player-1 states S_1 and player-2 states S_2 (i.e., $S = S_1 \cup S_2$ and $S_1 \cap S_2 = \emptyset$),
- $\rightarrow \subseteq S \times \mathbb{R}^+ \times S$ is the transition relation such that for all $s \in S$ there exists at least one $s' \in S$ such that for some Δ , we have $s \xrightarrow{\Delta} s'$.

Plays, objectives, strategies, outcomes *etc.* are as in finite games (Section 4).

Quantitative Timed Simulation Functions. Analogous to the game theoretic presentation of timed simulation games, we now present a game theoretic definition of quantitative timed simulation functions. Recall the two player turn-based bipartite timed transition game structure $\mathfrak{S}_t(A_r, A_s)$ defined in Section 3. Consider a play ρ in $\mathfrak{S}_t(A_r, A_s)$:

$$\langle s_r^0, s_s^0, 2 \rangle \xrightarrow{\Delta_r^0} \langle s_r^1, s_s^0, 1 \rangle \xrightarrow{\Delta_s^0} \langle s_r^1, s_s^1, 2 \rangle \xrightarrow{\Delta_r^1} \dots$$

Let $\rho(r)$ be the projection on A_r , thus $\rho(r)$ is the A_r trajectory $s_r^0 \xrightarrow{\Delta_r^0} s_r^1 \xrightarrow{\Delta_r^1} \dots$. Note that $\rho(r)$ is a valid trajectory in A_r . We define $\rho(s)$ similarly.

Definition 1 (Metric Over Simulation Game Plays). Recall the $\mathcal{D}_{\text{MaxDiff}}$, $\mathcal{D}_{\text{LimMaxDiff}}$, $\mathcal{D}_{\text{AvgDiff}}$ trajectory trace difference metrics defined in Section 2. For $\Psi \in \{\mathcal{D}_{\text{MaxDiff}}, \mathcal{D}_{\text{LimMaxDiff}}, \mathcal{D}_{\text{AvgDiff}}\}$, we define $\Psi^{\text{Timediv}}()$ as follows for a play ρ in $\mathfrak{S}_t(A_r, A_s)$:

$$\Psi^{\text{Timediv}}(\rho) = \begin{cases} 0 & \text{if } \rho(r) \notin \text{Timediv}(A_r) \\ \Psi(\rho(r), \rho(s)) & \text{otherwise} \end{cases} \quad \square$$

Note that Ψ^{Timediv} can be viewed as a metric over trajectories of timed transition systems.

Definition 2 (Quantitative Timed Simulation Functions). Let A_r, A_s be timed transition systems, and let $\mathfrak{S}_t(A_r, A_s)$ be the two player turn-based bipartite timed transition game structure defined in Section 3. The value of the quantitative simulation function $\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle s_r, s_s \rangle)$, for s_r and s_s states of A_r and A_s respectively, and for $\Psi^{\text{Timediv}} \in \{\mathcal{D}_{\text{MaxDiff}}^{\text{Timediv}}, \mathcal{D}_{\text{LimMaxDiff}}^{\text{Timediv}}, \mathcal{D}_{\text{AvgDiff}}^{\text{Timediv}}\}$, is defined as follows.

$$\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle s_r, s_s \rangle) = \inf_{\pi_s \in \Pi_s} \sup_{\pi_r \in \Pi_r} \Psi^{\text{Timediv}}(\rho(\pi_r, \pi_s, \langle s_r, s_s, 2 \rangle))$$

where $\rho(\pi_r, \pi_s, \langle s_r, s_s, 2 \rangle)$ is the trajectory which results given the player-1 strategy $\pi_s \in \Pi_s$ and the player-2 strategy $\pi_r \in \Pi_r$. Equivalently,

$$\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle s_r, s_s \rangle) = \text{Opt}(\Psi^{\text{Timediv}})(\langle s_r, s_s, 2 \rangle) \quad \square$$

The following lemma shows that $\mathcal{S}_{\mathcal{D}_{\text{MaxDiff}}^{\text{Timediv}}}$ and $\mathcal{S}_{\mathcal{D}_{\text{MaxDiff}}}$ coincide for well-formed TTS (the function $\mathcal{S}_{\mathcal{D}_{\text{MaxDiff}}}$ is defined without regard for time-divergence, i.e. using $\Psi = \mathcal{D}_{\text{MaxDiff}}$ instead of Ψ^{Timediv} in Definition 2).

Proposition 7. Let A_r and A_s be timed transition systems, and let A_r be well-formed. For any states s_r of A_r and s_s of A_s , we have

$$\mathcal{S}_{\mathcal{D}_{\text{MaxDiff}}^{\text{Timediv}}}(\langle s_r, s_s \rangle) = \mathcal{S}_{\mathcal{D}_{\text{MaxDiff}}}(\langle s_r, s_s \rangle) \quad \square$$

5.2 Timed Automata

Timed automata [AD94] suggest a finite syntax for specifying infinite-state timed game structures. A *timed automaton* \mathcal{T} is a tuple $\langle L, \Sigma, C, \mu, \rightarrow, \gamma, S_0 \rangle$, where

- L is the set of locations.
- Σ is the set of atomic propositions.
- C is a finite set of clocks. A *clock valuation* $v : C \mapsto \mathbb{R}^+$ for a set of clocks C assigns a real value to each clock in C .
- $\mu : L \mapsto 2^\Sigma$ is the observation map (it does not depend on clock values).
- $\rightarrow \subseteq L \times L \times 2^C \times \Phi(C)$ gives the set of transitions, where $\Phi(C)$ is the set of clock constraints generated by $\psi := x \leq d \mid d \leq x \mid \neg\psi \mid \psi_1 \wedge \psi_2$.
- $\gamma : L \mapsto \text{Constr}(C)$ is a function that assigns to every location an invariant on clock valuations. All clocks increase uniformly at the same rate. When at location l , a valid execution must move out of l before the invariant $\gamma(l)$ expires. Thus, the timed automaton can stay at a location only as long as the invariant is satisfied by the clock values.
- $S_0 \subseteq L \times \mathbb{R}^{|C|}$ is the set of initial states.

Each clock increases at rate 1 inside a location. A *clock valuation* is a function $\kappa : C \mapsto \mathbb{R}_{\geq 0}$ that maps every clock to a nonnegative real. The set of all clock valuations for C is denoted by $K(C)$. Given a clock valuation $\kappa \in K(C)$ and a time delay $\Delta \in \mathbb{R}_{\geq 0}$, we write $\kappa + \Delta$ for the clock valuation in $K(C)$ defined by $(\kappa + \Delta)(x) = \kappa(x) + \Delta$ for all clocks $x \in C$. For a subset $\lambda \subseteq C$ of the clocks, we write $\kappa[\lambda := 0]$ for the clock valuation in $K(C)$ defined by $(\kappa[\lambda := 0])(x) = 0$ if $x \in \lambda$, and $(\kappa[\lambda := 0])(x) = \kappa(x)$ if $x \notin \lambda$. A clock valuation $\kappa \in K(C)$ *satisfies* the clock constraint θ , written $\kappa \models \theta$, if the condition θ holds when all clocks in C take on the values specified by κ . A *state* $s = \langle l, \kappa \rangle$ of the timed automaton \mathcal{T} is a location $l \in L$ together with a clock valuation $\kappa \in K(C)$ such that the invariant at the location is satisfied, that is, $\kappa \models \gamma(l)$. We let S be the set of all states of \mathcal{T} . An edge $\langle l, l', \lambda, \theta \rangle$ represents a transition from location l to location l' when the clock values at l satisfy the constraint θ . The set $\lambda \subseteq C$ gives the clocks to be reset to 0 with this transition. The semantics of timed automata are given as timed transition systems. This is standard [AD94], and omitted here.

Clock Region Equivalence. Clock region equivalence, denoted as \cong is an equivalence relation on states of timed automata. The equivalence classes of the relation are called *regions*, and induce a time abstract bisimulation on the corresponding timed transition system. There are finitely many clock regions; more precisely, the number of clock regions is bounded by $|L| \cdot \prod_{x \in C} (c_x + 1) \cdot |C|! \cdot 4^{|C|}$. For a real $t \geq 0$, let $\text{frac}(t) = t - \lfloor t \rfloor$ denote the fractional part of t . Given a timed automaton game \mathcal{T} , for each clock $x \in C$, let c_x denote the largest integer constant that appears in any clock constraint involving x in \mathcal{T} (let $c_x = 1$ if there is no clock constraint involving x). Two states $\langle l_1, \kappa_1 \rangle$ and $\langle l_2, \kappa_2 \rangle$ are said to be *region equivalent* if all the following conditions are satisfied: (a) $l_1 = l_2$, (b) for all clocks x , $\kappa_1(x) \leq c_x$ iff $\kappa_2(x) \leq c_x$, (c) for all clocks x with $\kappa_1(x) \leq c_x$, $\lfloor \kappa_1(x) \rfloor = \lfloor \kappa_2(x) \rfloor$, (d) for all clocks x, y with $\kappa_1(x) \leq c_x$ and $\kappa_1(y) \leq c_y$, $\text{frac}(\kappa_1(x)) \leq \text{frac}(\kappa_1(y))$ iff $\text{frac}(\kappa_2(x)) \leq \text{frac}(\kappa_2(y))$, and (e) for all clocks x with $\kappa_1(x) \leq c_x$, $\text{frac}(\kappa_1(x)) = 0$ iff $\text{frac}(\kappa_2(x)) = 0$. Given a state $\langle l, \kappa \rangle$ of \mathcal{T} , we denote the region containing $\langle l, \kappa \rangle$ as $\text{Reg}(\langle l, \kappa \rangle)$.

Region Graph. The region graph $\text{Reg}(\mathcal{T})$ corresponding to \mathcal{T} is the time-abstract bisimulation quotient graph induced by the region equivalence relation. The states of $\text{Reg}(\mathcal{T})$ are the regions of \mathcal{T} . There is a transition $R \rightarrow R'$ iff there exists $s \in R$ and $s' \in R'$ such that $s \xrightarrow{\Delta} s'$ for some $\Delta \geq 0$.

5.3 Computation of Quantitative Simulation Functions on Timed Automata

In this subsection we solve for quantitative simulation functions on timed automata by reducing the problem to games on finite-state graphs. For ease of presentation we assume that all clocks are bounded, *i.e.*, that the invariants of each location can be conjuncted with the clause $\bigwedge_{x \in C} (x \leq c_{\max})$ for some constant c_{\max} . The general case where clocks may be unbounded can be solved using similar algorithms, with some additional bookkeeping.

The solution involves the following steps. We first enlarge the timed game structure corresponding to \mathcal{T} in Sub-subsection 5.3 in order to measure elapsed time, and to measure the integer time boundaries crossed. Then, we define *integer* quantitative simulation functions which depend only on the integer time boundaries crossed in Sub-subsection 5.3, and show that these integer simulation functions are close to the original (real-valued) simulation functions. Then, we show that these integer simulation functions can be computed on finite game graphs in Sub-subsection 5.3. Finally, we present the algorithm which ties all the steps together, and show that we can compute the quantitative simulation functions to within any desired degree of accuracy.

Enlarging the Timed Game Structure Given a timed automata \mathcal{T} where all the clocks are bounded by c_{\max} , let $\llbracket \mathcal{T} \rrbracket$ denote the timed transition system obtained by adding to \mathcal{T} an extra clock z , which cycles between 0 and 1 for measuring elapsed time, and an integer valued variable *ticks* which takes on values in $\mathbb{N}_{\leq c_{\max}}$, where $\mathbb{N}_{\leq c_{\max}}$ denotes the set $\{0, 1, \dots, c_{\max}\}$. Formally, the set of states of $\llbracket \mathcal{T} \rrbracket$ is $S^{\llbracket \mathcal{T} \rrbracket} = S \times \mathbb{R}_{[0,1]} \times \mathbb{N}_{\leq c_{\max}}$, where S is the set of states of \mathcal{T} . The state $\langle s, \mathfrak{z}, \mathfrak{d} \rangle$ of $\llbracket \mathcal{T} \rrbracket$ has the following components:

- s is the state of the original timed automaton \mathcal{T} ;
- \mathfrak{z} is the value of the added clock z which gets reset to 0 every time it crosses 1 (*i.e.*, if κ' is the clock valuation resulting from letting time Δ elapse from an initial clock valuation κ , then, $\mathfrak{z} = \kappa'(z) = (\kappa(z) + \Delta) \bmod 1$); and
- \mathfrak{d} denotes the value of the integer variable *ticks*, and is equal to the number of integer boundaries crossed by the added clock z since the last transition: if the clock valuation in the previous state was κ , and the transition time duration is Δ , then $\mathfrak{d} = \lfloor \kappa(z) + \Delta \rfloor$ in the current state, where $\lfloor \cdot \rfloor$ denotes the integer floor function. Note that since all the clocks in \mathcal{T} are bounded by c_{\max} , we have $\mathfrak{d} \leq c_{\max}$, as the maximum duration of a transition is c_{\max} , and $\kappa(z) < 1$ in the previous state.

The region equivalence relation can be expanded to $\llbracket \mathcal{T} \rrbracket$ states. Two states $\langle \langle l_1, \kappa_1 \rangle, \mathfrak{z}_1, \mathfrak{d}_1 \rangle$ and $\langle \langle l_2, \kappa_2 \rangle, \mathfrak{z}_2, \mathfrak{d}_2 \rangle$ of $\llbracket \mathcal{T} \rrbracket$ are defined to be region equivalent if $\langle l_1, \mathfrak{z}_1 \rangle = \langle l_2, \mathfrak{z}_2 \rangle$, and $\kappa_1^{z=\mathfrak{z}_1} \cong \kappa_2^{z=\mathfrak{z}_2}$, where $\kappa_i^{z=\mathfrak{z}_i}$ denotes the clock valuation κ_i on C expanded to a clock valuation to $C \cup \{z\}$ by mapping z to \mathfrak{z}_i (we denote the enlarged clock valuation be denoted as $\hat{\kappa}$). Similar to the region graph $\text{Reg}(\mathcal{T})$, we define an untimed finite state bisimulation quotient graph $\text{Reg}(\llbracket \mathcal{T} \rrbracket)$ for $\llbracket \mathcal{T} \rrbracket$.

Given a state s of \mathcal{T} , we denote by $\llbracket s \rrbracket$ the state $\langle s, 0, 0 \rangle$ of $\llbracket \mathcal{T} \rrbracket$. For a state trajectory $\text{traj} = s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} \dots$, we let $\text{traj}[i]$ denote the state s_i . Given a state trajectory traj of the timed automaton \mathcal{T} , we denote by $\llbracket \text{traj} \rrbracket$ the $\llbracket \mathcal{T} \rrbracket$ trajectory $\llbracket \text{traj}[0] \rrbracket \xrightarrow{t_0} \hat{s}_1 \xrightarrow{t_1} \hat{s}_2 \dots$, where $\hat{s}_i = \langle s_i, \mathfrak{z}_i, \mathfrak{d}_i \rangle$, and $\mathfrak{z}_i, \mathfrak{d}_i$ values are according to the times of the transitions (letting $\llbracket \text{traj}[0] \rrbracket = \hat{s}_0$). That is, $\llbracket \text{traj} \rrbracket$ denotes the trajectory obtained by adding the clock z , and the integer variable *ticks*, where the values for both the new variables are set to 0 in the starting state $\llbracket \text{traj}[0] \rrbracket$. The new variables just observe the time, and the integer boundaries crossed for each transition according to the semantics for $\llbracket \mathcal{T} \rrbracket$ described previously. The first component of $\llbracket \text{traj}[i] \rrbracket$ is the same as the state $\text{traj}[i]$ for all i .

The next Lemma shows that a trajectory is time-divergent iff it satisfies a Büchi constraint.

Lemma 6. *Let traj be a trajectory of a timed automaton \mathcal{T} in which all clocks are bounded by M . The trajectory traj is time-divergent iff $\llbracket \text{traj} \rrbracket$ satisfies the Büchi condition $\text{Büchi} \left(\bigvee_{i=1}^M \text{ticks} = i \right)$*

Proof. The proof follows from the fact that trajectory traj is *not* time-divergent iff global time does not progress beyond some integer U . This happens iff time crosses only finitely many integer boundaries. Now, global time crosses an integer boundary at step n iff $\left(\bigvee_{i=1}^M \text{ticks} = i \right)$ is true at step n . Thus trajectory traj is *not* time-divergent iff $\left(\bigvee_{i=1}^M \text{ticks} = i \right)$ is true only finitely often. Equivalently, trajectory traj is time-divergent iff $\left(\bigvee_{i=1}^M \text{ticks} = i \right)$ is true infinitely often. \square

Integer Time. For the trajectory $\llbracket \text{traj} \rrbracket$, let $\text{time}_{\llbracket \text{traj} \rrbracket}^{\text{int}}[i]$ denote the number of integer boundaries crossed upto the i -th transition:

$$\text{time}_{\llbracket \text{traj} \rrbracket}^{\text{int}}[i] = \lfloor \text{time}_{\llbracket \text{traj} \rrbracket}[i] \rfloor$$

We have the following lemma which expresses $\text{time}_{\llbracket \text{traj} \rrbracket}^{\text{int}}[i]$ in terms of the values of the ticks variable in traces. Note that the value of the ticks variable is zero in the first state of a valid trajectory $\llbracket \text{traj} \rrbracket$.

Lemma 7. *Let traj be a trajectory of a timed automaton \mathcal{T} in which all clocks are bounded. We have*

$$\text{time}_{\llbracket \text{traj} \rrbracket}^{\text{int}}[i] = \sum_{j=0}^i d_j$$

where d_i is the value of the ticks variable in $\llbracket \text{traj} \rrbracket[i]$.

Proof. The proof follows from the definition of the ticks variable updates: the updates count the integer boundaries crossed by the z which measures elapsed time \square

The Integer Trace Difference Metrics $\mathcal{D}_{\text{MaxDiff}}^{\text{int}}$, $\mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}$, and $\mathcal{D}_{\text{AvgDiff}}^{\text{int}}$ Corresponding to the trace difference metric $\mathcal{D}_{\varphi}()$, for $\varphi = \text{MaxDiff}, \text{LimMaxDiff}, \text{AvgDiff}$, we define the trace difference metric $\mathcal{D}_{\varphi}^{\text{int}}()$, by substituting $\text{time}^{\text{int}}()$ for $\text{time}()$ in the definition of $\mathcal{D}_{\varphi}()$, and using only the location component of \mathcal{T} for matching. *E.g.*, letting $\llbracket \text{traj} \rrbracket[n] = \langle \langle l_n, \kappa_n \rangle, \mathfrak{z}_n, d_n \rangle$ and $\llbracket \text{traj}' \rrbracket[n] = \langle \langle l'_n, \kappa'_n \rangle, \mathfrak{z}'_n, d'_n \rangle$, we have $\mathcal{D}_{\text{MaxDiff}}^{\text{int}}(\llbracket \text{traj} \rrbracket, \llbracket \text{traj}' \rrbracket) =$

$$\begin{cases} \infty & \text{if } \mu(l_n) \neq \mu(l'_n) \\ & \text{for some } n \\ \sup_n \{ |\text{time}_{\llbracket \text{traj} \rrbracket}^{\text{int}}(n) - \text{time}_{\llbracket \text{traj}' \rrbracket}^{\text{int}}(n)| \} & \text{otherwise} \end{cases}$$

The following Lemma shows that $\mathcal{D}_{\varphi}^{\text{int}}()$ closely approximates $\mathcal{D}_{\varphi}()$.

Lemma 8. *Let traj_1 and traj_2 be two trajectories of a timed automaton \mathcal{T} . The following assertions are true for $\varphi \in \{\text{MaxDiff}, \text{LimMaxDiff}, \text{AvgDiff}\}$.*

1. $\mathcal{D}_{\varphi}(\llbracket \text{traj}_1 \rrbracket, \llbracket \text{traj}_2 \rrbracket) = \infty$ iff $\mathcal{D}_{\varphi}^{\text{int}}(\llbracket \text{traj}_1 \rrbracket, \llbracket \text{traj}_2 \rrbracket) = \infty$.

2. If both $\mathcal{D}_\varphi(\llbracket \text{traj}_1 \rrbracket, \llbracket \text{traj}_2 \rrbracket)$ and $\mathcal{D}_\varphi^{\text{int}}(\llbracket \text{traj}_1 \rrbracket, \llbracket \text{traj}_2 \rrbracket)$ are less than ∞ , then

$$\mathcal{D}_\varphi(\llbracket \text{traj}_1 \rrbracket, \llbracket \text{traj}_2 \rrbracket) + 1 \geq \mathcal{D}_\varphi^{\text{int}}(\llbracket \text{traj}_1 \rrbracket, \llbracket \text{traj}_2 \rrbracket) \geq \mathcal{D}_\varphi(\llbracket \text{traj}_1 \rrbracket, \llbracket \text{traj}_2 \rrbracket) - 1.$$

Proof. Let us denote the sequence $\text{time}_{\llbracket \text{traj} \rrbracket}(n)$ as $x(n)$, the sequence $\text{time}_{\llbracket \text{traj}' \rrbracket}(n)$ as $x'(n)$, the sequence $\text{time}_{\llbracket \text{traj} \rrbracket}^{\text{int}}(n)$ as $y(n)$ and the sequence $\text{time}_{\llbracket \text{traj}' \rrbracket}^{\text{int}}(n)$ as $y'(n)$. We have

$$\begin{aligned} x(n) - 1 &< y(n) < x(n) \\ x'(n) - 1 &< y'(n) < x'(n) \end{aligned}$$

Thus, we have

$$x(n) - x'(n) - 1 < y(n) - y'(n) < x(n) - x'(n) + 1$$

Hence

$$|x(n) - x'(n)| - 1 < |y(n) - y'(n)| < |x(n) - x'(n)| + 1$$

It follows that

$$\sup_n |x(n) - x'(n)| - 1 < \sup_n |y(n) - y'(n)| < \sup_n |x(n) - x'(n)| + 1$$

Thus, we have the results for $\varphi = \text{MaxDiff}$.

We also have the following two relationships

$$\begin{aligned} \lim_{U \rightarrow \infty} \sup_{n > U} |x(n) - x'(n)| - 1 &< \lim_{U \rightarrow \infty} \sup_{n > U} |y(n) - y'(n)| \\ \lim_{U \rightarrow \infty} \sup_{n > U} |y(n) - y'(n)| &< \lim_{U \rightarrow \infty} \sup_{n > U} |x(n) - x'(n)| + 1 \end{aligned}$$

This gives the results for $\varphi = \text{LimMaxDiff}$.

Next, we note that for every n , the following two relationships hold.

$$\begin{aligned} \frac{\sum_{j=0}^n x(j) - n}{n} &< \frac{\sum_{j=0}^n y(j)}{n} < \frac{\sum_{j=0}^n x'(j)}{n} \\ \frac{\sum_{j=0}^n x'(j) - n}{n} &< \frac{\sum_{j=0}^n y'(j)}{n} < \frac{\sum_{j=0}^n x(j)}{n} \end{aligned}$$

And thus,

$$\begin{aligned} \frac{\sum_{j=0}^n x(j)}{n} - 1 &< \frac{\sum_{j=0}^n y(j)}{n} < \frac{\sum_{j=0}^n x'(j)}{n} \\ \frac{\sum_{j=0}^n x'(j)}{n} - 1 &< \frac{\sum_{j=0}^n y'(j)}{n} < \frac{\sum_{j=0}^n x(j)}{n} \end{aligned}$$

Then, applying similar reasoning as in $\varphi = \text{LimMaxDiff}$, we get the results for $\varphi = \text{AvgDiff}$. \square

Integer Quantitative Timed Simulation Functions Using $\mathcal{D}_{\text{MaxDiff}}^{\text{int}}$, $\mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}$, and $\mathcal{D}_{\text{AvgDiff}}^{\text{int}}$, we can define integer quantitative simulation functions which approximate $\mathcal{S}_{\Psi^{\text{Timediv}}}$ for $\Psi^{\text{Timediv}} \in \{\mathcal{D}_{\text{MaxDiff}}^{\text{Timediv}}, \mathcal{D}_{\text{LimMaxDiff}}^{\text{Timediv}}, \mathcal{D}_{\text{AvgDiff}}^{\text{Timediv}}\}$.

Definition 3 (Integer Metric over Simulation Game Plays). For $\Lambda \in \{\mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \mathcal{D}_{\text{AvgDiff}}^{\text{int}}\}$, we define $\Lambda^{\text{Timediv}}()$ as follows for a play ρ in $\mathfrak{S}_t(\llbracket \mathcal{T}_{\mathbf{r}} \rrbracket, \llbracket \mathcal{T}_{\mathbf{s}} \rrbracket)$:

$$\Lambda^{\text{Timediv}}(\rho) = \begin{cases} 0 & \text{if } \rho(\mathbf{r}) \notin \text{Timediv}(\llbracket \mathcal{T}_{\mathbf{r}} \rrbracket) \\ \Lambda(\rho(\mathbf{r}), \rho(\mathbf{s})) & \text{otherwise} \end{cases} \quad \square$$

The integer quantitative simulation functions $\mathcal{S}_{\Lambda^{\text{Timediv}}}(\langle s_{\mathbf{r}}, s_{\mathbf{s}} \rangle)$, can now be defined exactly as in Definition 2, using Λ^{Timediv} instead of Ψ^{Timediv} . The formal definition is given below in Definition 4

Definition 4 (Integer Quantitative Timed Simulation Functions). Let $\mathcal{T}_{\mathbf{r}}, \mathcal{T}_{\mathbf{s}}$ be timed automata, with the corresponding enlarged timed transition systems $\llbracket \mathcal{T}_{\mathbf{r}} \rrbracket, \llbracket \mathcal{T}_{\mathbf{s}} \rrbracket$ respectively, and let $\mathfrak{S}_t(\llbracket A_{\mathbf{r}} \rrbracket, \llbracket A_{\mathbf{s}} \rrbracket)$ be the two player turn-based bipartite timed simulation game structure. The value of the integer quantitative simulation function $\mathcal{S}_{\Lambda^{\text{Timediv}}}(\langle \llbracket s_{\mathbf{r}} \rrbracket, \llbracket s_{\mathbf{s}} \rrbracket \rangle)$, for $\llbracket s_{\mathbf{r}} \rrbracket$ and $\llbracket s_{\mathbf{s}} \rrbracket$ states of $\llbracket A_{\mathbf{r}} \rrbracket$ and $\llbracket A_{\mathbf{s}} \rrbracket$ respectively, and for $\Lambda \in \{\mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \mathcal{D}_{\text{AvgDiff}}^{\text{int}}\}$, is defined as follows.

$$\mathcal{S}_{\Lambda^{\text{Timediv}}}(\langle \llbracket s_{\mathbf{r}} \rrbracket, \llbracket s_{\mathbf{s}} \rrbracket \rangle) = \inf_{\pi_{\mathbf{s}} \in \Pi_{\mathbf{s}}} \sup_{\pi_{\mathbf{r}} \in \Pi_{\mathbf{r}}} \Lambda^{\text{Timediv}}(\rho(\pi_{\mathbf{r}}, \pi_{\mathbf{s}}, \langle \llbracket s_{\mathbf{r}} \rrbracket, \llbracket s_{\mathbf{s}} \rrbracket, 2 \rangle))$$

where $\rho(\pi_{\mathbf{r}}, \pi_{\mathbf{s}}, \langle \llbracket s_{\mathbf{r}} \rrbracket, \llbracket s_{\mathbf{s}} \rrbracket, 2 \rangle)$ is the trajectory which results given the player-1 strategy $\pi_{\mathbf{s}} \in \Pi_{\mathbf{s}}$ and the player-2 strategy $\pi_{\mathbf{r}} \in \Pi_{\mathbf{r}}$. \square

Let ρ be a play of the simulation game $\mathfrak{S}_t(\llbracket A_{\mathbf{r}} \rrbracket, \llbracket A_{\mathbf{s}} \rrbracket)$. The next lemma states that closeness of the trajectories $\rho(\mathbf{r}), \rho(\mathbf{s})$ according to integer trajectory distances is approximately the same as the normal (real-valued) trajectory distances,

Lemma 9. Let $\mathcal{T}_{\mathbf{r}}, \mathcal{T}_{\mathbf{s}}$ be timed automata, with the corresponding enlarged timed transition systems $\llbracket \mathcal{T}_{\mathbf{r}} \rrbracket, \llbracket \mathcal{T}_{\mathbf{s}} \rrbracket$ respectively, and let $\mathfrak{S}_t(\llbracket A_{\mathbf{r}} \rrbracket, \llbracket A_{\mathbf{s}} \rrbracket)$ be the two player turn-based bipartite timed simulation game structure. The following assertions are true for $\langle \Lambda, \Psi \rangle \in \{\langle \mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \mathcal{D}_{\text{MaxDiff}} \rangle, \langle \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \mathcal{D}_{\text{LimMaxDiff}} \rangle, \langle \mathcal{D}_{\text{AvgDiff}}^{\text{int}}, \mathcal{D}_{\text{AvgDiff}} \rangle\}$, for any play ρ of $\mathfrak{S}_t(\llbracket A_{\mathbf{r}} \rrbracket, \llbracket A_{\mathbf{s}} \rrbracket)$.

1. $\Lambda^{\text{Timediv}}(\rho) = \infty$ iff $\Psi^{\text{Timediv}}(\rho) = \infty$.
2. If both $\Lambda^{\text{Timediv}}(\rho)$ and $\Psi^{\text{Timediv}}(\rho)$ are less than ∞ then

$$\left| \Lambda^{\text{Timediv}}(\rho) - \Psi^{\text{Timediv}}(\rho) \right| \leq 1$$

Proof. The result follows from Lemma 8 and by the definitions of $\Lambda^{\text{Timediv}}(\rho)$ and $\Psi^{\text{Timediv}}(\rho)$. \square

Lemma 10. Let $\{\langle x_{r,s}, y_{r,s} \rangle \mid r \in R, s \in S\}$ be a set of tuples of numbers for some give sets R, S such that $x_{r,s} \in \mathbb{R}_{\infty}^+$ and $y_{r,s} \in \mathbb{R}_{\infty}^+$ where $\mathbb{R}_{\infty}^+ = \mathbb{R}^+ \cup \{\infty\}$. Let both the following conditions hold:

1. For all r, s we have $x_{r,s} = \infty$ iff $y_{r,s} = \infty$.
2. There exists some $\alpha \in \mathbb{R}^+$ such that for all r, s , if
 - $x_{r,s} \neq \infty$ and
 - $y_{r,s} \neq \infty$
then, $|x_{r,s} - y_{r,s}| \leq \alpha$.

Then, the following assertion are true.

1. $\inf_{s \in S} \sup_{r \in R} x_{r,s} = \infty$ iff $\inf_{s \in S} \sup_{r \in R} y_{r,s} = \infty$.
2. If $\inf_{s \in S} \sup_{r \in R} x_{r,s} < \infty$ and $\inf_{s \in S} \sup_{r \in R} y_{r,s} < \infty$ then

$$\left| \inf_{s \in S} \sup_{r \in R} x_{r,s} - \inf_{s \in S} \sup_{r \in R} y_{r,s} \right| \leq \alpha$$

Proof. We prove both the assertions.

1. Suppose $\inf_{s \in S} \sup_{r \in R} x_{r,s} = \infty$ (the other direction is symmetric). We must have that for every $s \in S$ the entity $\sup_{r \in R} x_{r,s} = \infty$. We show that:

Fact-1: For every $s \in S$, if $\sup_{r \in R} x_{r,s} = \infty$, the entity $\sup_{r \in R} y_{r,s} = \infty$.

Fix some $s \in S$.

- If there exists some $r \in R$ such that $x_{r,s} = \infty$, then by the conditions of the lemma, $y_{r,s} = \infty$. Thus $\sup_{r \in R} y_{r,s} = \infty$.
- Suppose for all $r \in R$ we have $x_{r,s} < \infty$. By the conditions of the lemma, for all $r \in R$ we have $|x_{r,s} - y_{r,s}| \leq \alpha$. Thus, if $\sup_{r \in R} x_{r,s} = \infty$, then $\sup_{r \in R} y_{r,s} = \infty$.

Thus **Fact-1** is true. Hence, $\sup_{r \in R} y_{r,s} = \infty$ for every $s \in S$. Thus, $\inf_{s \in S} \sup_{r \in R} y_{r,s} = \infty$.

2. Suppose we have both $\inf_{s \in S} \sup_{r \in R} x_{r,s} < \infty$ and $\inf_{s \in S} \sup_{r \in R} y_{r,s} < \infty$.

Fix some $s \in S$.

- Suppose $\sup_{r \in R} x_{r,s} = \infty$. We have that $\sup_{r \in R} y_{r,s} = \infty$ by **Fact-1** above.
- Suppose $\sup_{r \in R} x_{r,s} < \infty$ (note that there must exist at least one such s otherwise $\inf_{s \in S} \sup_{r \in R} x_{r,s} = \infty$). Thus, for this s , we have that for all $r \in R$, the quantity $x_{r,s} < \infty$. By the conditions of the lemma, we have that for this s , for all $r \in R$, the quantity $y_{r,s} < \infty$, and that $|x_{r,s} - y_{r,s}| \leq \alpha$. This implies that

$$\left| \sup_{r \in R} x_{r,s} - \sup_{r \in R} y_{r,s} \right| \leq \alpha$$

Let $p_s = \sup_{r \in R} x_{r,s}$, and $q_s = \sup_{r \in R} y_{r,s}$. From above, we have that for all s , it holds that either

- $p_s = q_s = \infty$, or
- $|p_s - q_s| \leq \alpha$.

Also, it holds that for at least one s , we have $p_s < \infty$. Thus, can throw away the p_s numbers such that $p_s < \infty$ in the computation of $\inf_s p_s$. For the rest, since $|p_s - q_s| \leq \alpha$, we have that $|\inf_s p_s - \inf_s q_s| \leq \alpha$. Thus, the second part of the assertion is true. \square

The following proposition states that the integer simulation functions closely approximate the original quantitative simulation functions.

Proposition 8 (Integer Simulation Functions Approximate Quantitative Simulation Functions). *Let $\mathcal{T}_r, \mathcal{T}_s$ be timed automata, with the corresponding enlarged timed transition systems $\llbracket \mathcal{T}_r \rrbracket, \llbracket \mathcal{T}_s \rrbracket$ respectively, and let $\mathfrak{S}_t(\llbracket A_r \rrbracket, \llbracket A_s \rrbracket)$ be the two player turn-based bipartite timed simulation game structure. For $\langle \Lambda, \Psi \rangle$ in $\{\langle \mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \mathcal{D}_{\text{MaxDiff}} \rangle, \langle \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \mathcal{D}_{\text{LimMaxDiff}} \rangle, \langle \mathcal{D}_{\text{AvgDiff}}^{\text{int}}, \mathcal{D}_{\text{AvgDiff}} \rangle\}$, we have the following assertions to be true.*

1. $\mathcal{S}_{\Lambda^{\text{Timediv}}}(\langle \llbracket s_r \rrbracket, \llbracket s_s \rrbracket \rangle) = \infty$ iff $\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle \llbracket s_r \rrbracket, \llbracket s_s \rrbracket \rangle) = \infty$.
2. If $\mathcal{S}_{\Lambda^{\text{Timediv}}}(\langle \llbracket s_r \rrbracket, \llbracket s_s \rrbracket \rangle) < \infty$ and $\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle \llbracket s_r \rrbracket, \llbracket s_s \rrbracket \rangle) < \infty$, then

$$|\mathcal{S}_{\Lambda^{\text{Timediv}}}(\langle \llbracket s_r \rrbracket, \llbracket s_s \rrbracket \rangle) - \mathcal{S}_{\Psi^{\text{Timediv}}}(\langle \llbracket s_r \rrbracket, \llbracket s_s \rrbracket \rangle)| \leq 1$$

Proof. The proof follows from Lemma 10 and Lemma 9. \square

Reduction to Games on Weighted Game Graphs In this section we show how to compute the values of the integer quantitative simulation functions by reductions to finite state games. First, we show that the values of the integer quantitative simulation functions are exactly the same on region graphs as on timed automata.

The Integer Trace Difference Metrics and Simulation Functions on Untimed Region Graphs. We first lift the integer trace difference metrics Λ^{Timediv} for $\Lambda \in \{\mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \mathcal{D}_{\text{AvgDiff}}^{\text{int}}\}$ to (untimed) region graphs. Let $\text{Reg}(\llbracket \mathcal{T} \rrbracket)$ be the region graph corresponding to the enlarged time game structure $\llbracket \mathcal{T} \rrbracket$ as defined in Sub-subsection 5.3. Let the observation function μ be defined as $\mu(\langle l, \kappa, \mathfrak{z}, d \rangle) = \langle \mu(l), d \rangle$.

Given the two timed automata $\mathcal{T}_\tau, \mathcal{T}_s$, consider the *untimed* simulation game $\mathfrak{S}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$ defined to be the untimed simulation game $\mathfrak{S}_u(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$, but with the observation function $\mu^\dagger(\langle l, \kappa, \mathfrak{z}, d \rangle) = \mu(l)$. For a play ρ of \mathfrak{S}_u^\dagger , we define $\rho(\tau)$ and $\rho(s)$ as the projections on $\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket)$ and $\text{Reg}(\llbracket \mathcal{T}_s \rrbracket)$ respectively. For a region graph $\text{Reg}(\llbracket \mathcal{T} \rrbracket)$ we define $\text{Timediv}(\text{Reg}(\llbracket \mathcal{T} \rrbracket))$ as the set of runs satisfying the Büchi condition $\text{Büchi}(\bigvee_{i=1}^{c_{\text{max}}} \text{ticks} = i)$. By Lemma 6, this has the intended meaning of encoding time divergence. Next we define Λ^{Timediv} for $\Lambda \in \{\mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \mathcal{D}_{\text{AvgDiff}}^{\text{int}}\}$ on plays of $\mathfrak{S}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$ using Lemma 7 as defining $\text{time}_{\text{Reg}(\llbracket \mathcal{T} \rrbracket)}^{\text{int}}(i)$ in terms of the *ticks* variable. Finally, we define the integer simulation functions as in Definition 4.

The next lemma states that the values of the integer simulation functions of the region graphs are the same as that on timed automata. Note that region graphs are *untimed* structures.

Lemma 11. *Let $\mathcal{T}_\tau, \mathcal{T}_s$ be timed automata, and let $\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket)$ be region graphs of the corresponding enlarged timed game structures $\llbracket \mathcal{T}_\tau \rrbracket, \llbracket \mathcal{T}_s \rrbracket$ respectively. For any states $\llbracket s_\tau \rrbracket$ of $\llbracket \mathcal{T}_\tau \rrbracket$ and $\llbracket s_s \rrbracket$ of $\llbracket \mathcal{T}_s \rrbracket$, we have*

$$\begin{aligned} \mathcal{S}_{\Lambda^{\text{Timediv}}}^{\mathfrak{S}_t(\llbracket \mathcal{T}_\tau \rrbracket, \llbracket \mathcal{T}_s \rrbracket)}(\langle \llbracket s_\tau \rrbracket, \llbracket s_s \rrbracket \rangle) \\ = \\ \mathcal{S}_{\Lambda^{\text{Timediv}}}^{\mathfrak{S}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))}(\langle \text{Reg}(\llbracket s_\tau \rrbracket), \text{Reg}(\llbracket s_s \rrbracket) \rangle) \end{aligned}$$

where $\Lambda \in \{\mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \mathcal{D}_{\text{AvgDiff}}^{\text{int}}\}$.

Proof. For any timed automata \mathcal{T} , we have that $\text{Reg}(\llbracket \mathcal{T} \rrbracket)$ is a bisimulation quotient of $\llbracket \mathcal{T} \rrbracket$ for the observation function μ . Thus, given any play ρ of $\mathfrak{S}_t(\llbracket \mathcal{T}_\tau \rrbracket, \llbracket \mathcal{T}_s \rrbracket)$, there exists a play ρ_{Reg} of $\mathfrak{S}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$ such that $\rho_{\text{Reg}}(\tau)$ and $\rho_{\text{Reg}}(s)$ have the same (untimed) observation trace sequences as $\rho(\tau)$ and $\rho(s)$. The dual fact for any play ρ_{Reg} of $\mathfrak{S}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$ also holds due to the bisimulation.

Consider the definition of the simulation function $\mathcal{S}_{\Lambda^{\text{Timediv}}}$ on any game structure. Note that it depends only on the values of μ on the timed automata locations, and the values of the *ticks* variable in the trajectory states, that it, it depends only on the untimed observation trace sequences of the plays. Since these untimed observation trace sequences are the same for $\mathfrak{S}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$ and $\mathfrak{S}_t(\llbracket \mathcal{T}_\tau \rrbracket, \llbracket \mathcal{T}_s \rrbracket)$ from above, we have the desired result. \square

The weighted finite untimed game graph $\mathfrak{F}(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$.

Now we construct a finite weighted game graph $\mathfrak{F}(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$, on which we can use the algorithms of Section 4, to compute the values of the integer quantitative simulation function for $\mathfrak{S}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$. The game structure \mathfrak{F} is essentially the untimed simulation game \mathfrak{S}_u^\dagger over

the region graphs, where weights are assigned to transitions based on the *tick* values of the region states. Formally, $\mathfrak{F}(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$ (denoted \mathfrak{F} in short) is the tuple $\langle S^\mathfrak{F}, \rightarrow^\mathfrak{F}, w^\mathfrak{F} \rangle$, where

- $S^\mathfrak{F} = S_1^\mathfrak{F} \cup S_2^\mathfrak{F}$, and
 - ★ The set of player-2 states is $S_2^\mathfrak{F} = S^{\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket)} \times S^{\text{Reg}(\llbracket \mathcal{T}_s \rrbracket)} \times \{2\}$, where $S^{\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket)}$ is the set of states of $\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket)$, and $S^{\text{Reg}(\llbracket \mathcal{T}_s \rrbracket)}$ is the set of states of $\text{Reg}(\llbracket \mathcal{T}_s \rrbracket)$.
 - ★ The set of player-1 states is $S_1^\mathfrak{F} = S^{\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket)} \times S^{\text{Reg}(\llbracket \mathcal{T}_s \rrbracket)} \times \{1\}$.
- $\rightarrow^\mathfrak{F}$ is the set of edges where
 - ★ The player-2 transitions are:
$$\langle \text{Reg}(\langle l_\tau, \hat{\kappa}_\tau, d_\tau \rangle), \text{Reg}(\llbracket s_s \rrbracket), 2 \rangle \longrightarrow \langle \text{Reg}(\langle l'_\tau, \hat{\kappa}'_\tau, d'_\tau \rangle), \text{Reg}(\llbracket s_s \rrbracket), 1 \rangle,$$
such that $\text{Reg}(\langle l_\tau, \hat{\kappa}_\tau, d_\tau \rangle) \longrightarrow \text{Reg}(\langle l'_\tau, \hat{\kappa}'_\tau, d'_\tau \rangle)$ is a valid transition in $\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket)$.
 - ★ The player-1 transitions are:
$$\langle \text{Reg}(\llbracket s_\tau \rrbracket), \text{Reg}(\langle l_s, \hat{\kappa}_s, d_s \rangle), 1 \rangle \longrightarrow \langle \text{Reg}(\llbracket s_\tau \rrbracket), \text{Reg}(\langle l'_s, \hat{\kappa}'_s, d'_s \rangle), 2 \rangle,$$
such that
 1. $\text{Reg}(\langle l_s, \hat{\kappa}_s, d_s \rangle) \longrightarrow \text{Reg}(\langle l'_s, \hat{\kappa}'_s, d'_s \rangle)$ is a valid transition in $\text{Reg}(\llbracket \mathcal{T}_s \rrbracket)$; and
 2. $\mu^\dagger(\text{Reg}(\llbracket s_\tau \rrbracket)) = \mu^\dagger(\text{Reg}(\llbracket s'_s \rrbracket))$, that is, the observation on the (timed automaton) location of $\text{Reg}(s'_s)$ is the same as the observation on the location of $\text{Reg}(s_\tau)$.

If there is no outgoing transition from a player-1 state according to the above rules, we add a dummy transition to a sink state s_{sink} which we define to be such that the **Opt** value for player 1 is ∞ for all objectives from s_{sink} .

- The weight function $w^\mathfrak{F}$ is given as follows.
 - ★ $w^\mathfrak{F}(e_2) = 0$ for any edge e_2 originating from a player-2 state.
 - ★ $w^\mathfrak{F} \left(\begin{array}{c} \langle \text{Reg}(\langle l_\tau, \hat{\kappa}_\tau, d_\tau \rangle), \text{Reg}(\langle l_s, \hat{\kappa}_s, d_s \rangle), 1 \rangle \longrightarrow \\ \langle \text{Reg}(\langle l_\tau, \hat{\kappa}_\tau, d_\tau \rangle), \text{Reg}(\langle l'_s, \hat{\kappa}'_s, d'_s \rangle), 2 \rangle \end{array} \right)$ is the value $d_\tau - d'_s$.

We note that d'_τ is the number of integer boundaries crossed by the clock z in a transition to go from any state in $\text{Reg}(\langle l_\tau, \hat{\kappa}_\tau, d_\tau \rangle)$ to any state in $\text{Reg}(\langle l'_\tau, \hat{\kappa}'_\tau, d'_\tau \rangle)$, and similarly for $\text{intran}(d_s, d'_s)$. Thus, the weights $d_\tau - d'_s$ encode the difference of the integer boundaries crossed by the clock z in the region graphs $\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket)$ and $\text{Reg}(\llbracket \mathcal{T}_s \rrbracket)$.

The next lemma states that to compute the values of the integer quantitative simulation functions on the region graphs, we can use the objectives **DiffSumCB**, **EvDiffSumCB**, **AvDiffSumCB** on the weighted finite game $\mathfrak{F}(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$.

Lemma 12. *Let \mathcal{T}_τ and \mathcal{T}_s be well-formed timed automata such that all clocks are bounded by c_{\max} , and let $\mathfrak{F}(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$ be the weighted game structure corresponding to $\mathfrak{G}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))$, as described above. Fix the coBüchi objective $\text{coBüchi}(\text{ticks}_\tau = 0)$ in the following. For $\langle \Lambda, \Xi \rangle$ equal to $\langle \mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \text{DiffSumCB} \rangle$, or $\langle \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \text{EvDiffSumCB} \rangle$, or $\langle \mathcal{D}_{\text{AvgDiff}}^{\text{int}}, \text{AvDiffSumCB} \rangle$, we have*

$$\begin{aligned} & \mathcal{S}_{A^{\text{Timediv}}}^{\mathfrak{G}_u^\dagger(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket))} \left(\langle \text{Reg}(\llbracket s_\tau \rrbracket), \text{Reg}(\llbracket s_s \rrbracket) \rangle \right) \\ &= \\ & \left(\text{Opt}^\mathfrak{F}(\text{Reg}(\llbracket \mathcal{T}_\tau \rrbracket), \text{Reg}(\llbracket \mathcal{T}_s \rrbracket)) (\Xi) \right) \left(\langle \text{Reg}(\llbracket s_\tau \rrbracket), \text{Reg}(\llbracket s_s \rrbracket), 2 \rangle \right) \end{aligned}$$

Proof. Note that every finite play $\rho^{\mathfrak{G}_u^\dagger}$ of \mathfrak{G}_u^\dagger in which player 1 has not lost (in the untimed simulation game) corresponds to a finite play $\rho^\mathfrak{F}$ in \mathfrak{F} in which the sink location s_{sink} has not been visited, and similarly for the other direction. The move choices for both players are the same, apart from s_{sink} transitions. Conversely, any states $\text{Reg}(\llbracket s_\tau \rrbracket), \text{Reg}(\llbracket s_s \rrbracket)$ are not untimed similar in \mathfrak{G}_u^\dagger iff in the game

\mathfrak{F} , for every player-2 strategy, player 1 has a strategy which forces the play into the sink location and thus leads to an ∞ value for all the quantitative objectives.

Thus, consider the case where $\text{Reg}(\llbracket s_r \rrbracket), \text{Reg}(\llbracket s_s \rrbracket)$ are untimed similar in \mathfrak{S}_u^\dagger . Now, Timediv has been defined as Büchi ($\bigvee_{i=1}^{c_{\max}} \text{ticks} = i$) earlier, on the untimed region graphs. This Büchi condition is equivalent to $\neg \text{coBüchi}(\text{ticks} = 0)$. Thus, the condition $\rho^{\mathfrak{S}_u^\dagger}(\mathbf{r}) \notin \text{Timediv}$ holds iff $\rho^{\mathfrak{F}} \in \text{coBüchi}(\text{ticks}_r = 0)$ holds. Finally, we note that for any play $\rho^{\mathfrak{S}_u^\dagger}(\pi_r, \pi_s, \langle \text{Reg}(\llbracket s_r \rrbracket), \text{Reg}(\llbracket s_s \rrbracket), 2 \rangle)$, the corresponding play $\rho^{\mathfrak{F}}(\pi_r, \pi_s, \langle \text{Reg}(\llbracket s_r \rrbracket), \text{Reg}(\llbracket s_s \rrbracket), 2 \rangle)$ is such that

1. For every $i > 0$, we have

$$\text{time}_{(\rho^{\mathfrak{S}_u^\dagger})_{(\mathbf{r})}}^{\text{int}}[i] - \text{time}_{(\rho^{\mathfrak{S}_u^\dagger})_{(\mathbf{s})}}^{\text{int}}[i] = \sum_{j=1}^i w^{\mathfrak{F}}(\rho^{\mathfrak{F}}[2j-1] \rightarrow \rho^{\mathfrak{F}}[2j])$$

2. For every $i \geq 0$, we have

$$w^{\mathfrak{F}}(\rho^{\mathfrak{F}}[2i] \rightarrow \rho^{\mathfrak{F}}[2i+1]) = 0$$

The desired results follow. \square

Precision of the Integer Simulation Functions. Given a positive integer $\alpha \geq 1$, and a timed automaton \mathcal{T} , let $\alpha \cdot \mathcal{T}$ denote the timed automaton obtained from \mathcal{T} by multiplying every constant by α . Note that if clocks are bounded by c_{\max} in \mathcal{T} , then clocks are bounded by $\alpha \cdot c_{\max}$ in $\alpha \cdot \mathcal{T}$. The automaton $\alpha \cdot \mathcal{T}$ is just \mathcal{T} with a blown up timescale. One time unit in \mathcal{T} corresponds to α time units in $\alpha \cdot \mathcal{T}$. We let $\alpha \cdot \llbracket \mathcal{T} \rrbracket = \llbracket \alpha \cdot \mathcal{T} \rrbracket$, and

$$\alpha \cdot \langle l, \kappa, \mathfrak{z}, \mathfrak{d} \rangle = \langle l, \alpha \cdot \kappa, \text{frac}(\alpha \cdot \mathfrak{z}), \lfloor \alpha \cdot \mathfrak{z} \rfloor + \alpha \cdot \mathfrak{d} \rangle$$

where $\text{frac}(\beta)$ denotes the fractional part of β , i.e. $\beta - \lfloor \beta \rfloor$ for $\beta \geq 0$. Note that in $\alpha \cdot \llbracket \mathcal{T} \rrbracket$, the clock z still cycles from 0 to 1. Thus, we first blow up the time scale of \mathcal{T} to obtain $\alpha \cdot \mathcal{T}$, and then take the expanded game structure $\llbracket \alpha \cdot \mathcal{T} \rrbracket$.

Lemma 13. *Let $\mathcal{T}_r, \mathcal{T}_s$ be timed automata, with the corresponding enlarged timed transition systems $\llbracket \mathcal{T}_r \rrbracket, \llbracket \mathcal{T}_s \rrbracket$ respectively, and let $\mathfrak{S}_t(\llbracket A_r \rrbracket, \llbracket A_s \rrbracket)$ be the two player turn-based bipartite timed simulation game structure. For $\Psi \in \{\mathcal{D}_{\text{MaxDiff}}, \mathcal{D}_{\text{LimMaxDiff}}, \mathcal{D}_{\text{AvgDiff}}\}$, for any α a positive integer, and for any states $\llbracket s_r \rrbracket$ and $\llbracket s_s \rrbracket$ of $\llbracket \mathcal{T}_r \rrbracket$ and $\llbracket \mathcal{T}_s \rrbracket$ respectively, we have*

$$\alpha \cdot \mathcal{S}_{\Psi^{\text{Timediv}}}^{\llbracket \mathcal{T}_r \rrbracket, \llbracket \mathcal{T}_s \rrbracket}(\langle \llbracket s_r \rrbracket, \llbracket s_s \rrbracket \rangle) = \mathcal{S}_{\Psi^{\text{Timediv}}}^{\alpha \cdot \llbracket \mathcal{T}_r \rrbracket, \alpha \cdot \llbracket \mathcal{T}_s \rrbracket}(\langle \alpha \cdot \llbracket s_r \rrbracket, \alpha \cdot \llbracket s_s \rrbracket \rangle)$$

Proof. The proof follows from observing that the times in $\alpha \cdot \mathcal{T}$ are just the times in \mathcal{T} multiplied by α . \square

The following lemma states that integer simulation functions can approximate the real-valued simulation functions to within any desired degree of accuracy.

Proposition 9 (Integer Simulation Functions Approximate Quantitative Simulation Functions to Any Desired Degree). *Let $\mathcal{T}_r, \mathcal{T}_s$ be timed automata, with the corresponding enlarged timed transition systems $\llbracket \mathcal{T}_r \rrbracket, \llbracket \mathcal{T}_s \rrbracket$ respectively, and let $\mathfrak{S}_t(\llbracket A_r \rrbracket, \llbracket A_s \rrbracket)$ be the two player turn-based bipartite timed simulation game structure. For $\langle \Lambda, \Psi \rangle$ in $\{\langle \mathcal{D}_{\text{MaxDiff}}^{\text{int}}, \mathcal{D}_{\text{MaxDiff}} \rangle, \langle \mathcal{D}_{\text{LimMaxDiff}}^{\text{int}}, \mathcal{D}_{\text{LimMaxDiff}} \rangle, \langle \mathcal{D}_{\text{AvgDiff}}^{\text{int}}, \mathcal{D}_{\text{AvgDiff}} \rangle\}$, and for any positive integer $\alpha > 0$, we have the following assertions to be true.*

1. $\mathcal{S}_{A^{\text{Timediv}}}(\langle \alpha \cdot \llbracket s_{\mathfrak{r}} \rrbracket, \alpha \cdot \llbracket s_{\mathfrak{s}} \rrbracket \rangle) = \infty$ iff $\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle \llbracket s_{\mathfrak{r}} \rrbracket, \llbracket s_{\mathfrak{s}} \rrbracket \rangle) = \infty$.
2. If $\mathcal{S}_{A^{\text{Timediv}}}(\langle \alpha \cdot \llbracket s_{\mathfrak{r}} \rrbracket, \alpha \cdot \llbracket s_{\mathfrak{s}} \rrbracket \rangle) < \infty$ and $\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle \llbracket s_{\mathfrak{r}} \rrbracket, \llbracket s_{\mathfrak{s}} \rrbracket \rangle) < \infty$, then

$$\left| \alpha^{-1} \cdot \mathcal{S}_{A^{\text{Timediv}}}(\langle \alpha \cdot \llbracket s_{\mathfrak{r}} \rrbracket, \alpha \cdot \llbracket s_{\mathfrak{s}} \rrbracket \rangle) - \mathcal{S}_{\Psi^{\text{Timediv}}}(\langle \llbracket s_{\mathfrak{r}} \rrbracket, \llbracket s_{\mathfrak{s}} \rrbracket \rangle) \right| \leq \frac{1}{\alpha}$$

Proof. The proof follows from Lemma 13 and Proposition 8 applied to $\alpha \cdot \mathcal{T}_{\mathfrak{r}}$ and $\alpha \cdot \mathcal{T}_{\mathfrak{s}}$. \square

Final Algorithms and Results Applying Proposition 9, and Lemma 12, and the results of the previous section on games on finite state game graphs, we obtain the following Theorem which states that the values for the quantitative simulation functions $\mathcal{S}_{\Psi^{\text{Timediv}}}$ for $\Psi \in \{\mathcal{D}_{\text{MaxDiff}}, \mathcal{D}_{\text{LimMaxDiff}}, \mathcal{D}_{\text{AvgDiff}}\}$, can be computed to within any desired degree of accuracy using the algorithm in the function $h_{\Psi, \alpha}(s_{\mathfrak{r}}, s_{\mathfrak{s}})$.

Theorem 5. Let $\mathcal{T}_{\mathfrak{r}}$ and $\mathcal{T}_{\mathfrak{s}}$ be well-formed timed automata such that all clocks are bounded by c_{\max} , and let $\alpha \geq 1$ be a positive integer. For $\Psi \in \{\mathcal{D}_{\text{MaxDiff}}, \mathcal{D}_{\text{LimMaxDiff}}, \mathcal{D}_{\text{AvgDiff}}\}$, and for $\mathcal{S}_{\Psi^{\text{Timediv}}}$ the quantitative timed simulation function, the function $h_{\Psi, \alpha}()$ is such that for any states $s_{\mathfrak{r}}$ of $\mathcal{T}_{\mathfrak{r}}$ and $s_{\mathfrak{s}}$ of $\mathcal{T}_{\mathfrak{s}}$, either

1. $\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle s_{\mathfrak{r}}, s_{\mathfrak{s}} \rangle) = h_{\Psi, \alpha}(s_{\mathfrak{r}}, s_{\mathfrak{s}}) = \infty$; or
2. Both values are finite and

$$|\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle s_{\mathfrak{r}}, s_{\mathfrak{s}} \rangle) - h_{\Psi, \alpha}(s_{\mathfrak{r}}, s_{\mathfrak{s}})| \leq \frac{1}{\alpha}$$

Proof. The proof follows from Proposition 9 and Lemma 12. Since $\mathfrak{F}(\text{Reg}(\llbracket \alpha \cdot \mathcal{T}_{\mathfrak{r}} \rrbracket), \text{Reg}(\llbracket \alpha \cdot \mathcal{T}_{\mathfrak{s}} \rrbracket))$ is a finite weighted game graph, the value of $h_{\Psi, \alpha}(s_{\mathfrak{r}}, s_{\mathfrak{s}})$ can be computed using the algorithms of Section 4. \square

Concluding Remarks. We have presented algorithms for computing the various types of quantitative timed simulation function values (to any desired degree of accuracy) for timed automata. We note that the optimal player-1 strategies in the games required for quantitative timed simulation function values are also computable, and are witnesses to the quantitative simulation function values (dual to simulation relations witnessing the simulation decision problem).

References

- [AD94] R. Alur and D. L. Dill. “A Theory of Timed Automata.” In: *Theor. Comput. Sci.* 126.2 (1994), pp. 183–235.
- [AFS09] L. d. Alfaro, M. Faella, and M. Stoelinga. “Linear and Branching System Metrics”. In: *IEEE Trans. Software Eng.* 35.2 (2009), pp. 258–273.
- [ASG01] A. C. Antoulas, D. C. Sorensen, and S. Gugercin. “A survey of model reduction methods for large-scale systems”. In: *Contemporary Mathematics* 280 (2001), pp. 193–219.
- [Bou+11] P. Bouyer et al. “Measuring Permissiveness in Parity Games: Mean-Payoff Parity Games Revisited”. In: *ATVA*. LNCS 6996. Springer, 2011, pp. 135–149.
- [Bre+03] F. v. Breugel et al. “An Intrinsic Characterization of Approximate Probabilistic Bisimilarity”. In: *FoSSaCS*. LNCS 2620. Springer, 2003, pp. 200–215.

```

Input   : States  $s_{\tau}, s_{\mathfrak{s}}$  from  $\mathcal{T}_{\tau}, \mathcal{T}_{\mathfrak{s}}$  respectively;
            $\Psi \in \{\mathcal{D}_{\text{MaxDiff}}, \mathcal{D}_{\text{LimMaxDiff}}, \mathcal{D}_{\text{AvgDiff}}\};$ 
            $\alpha$  a positive integer
Output: A number approximating  $\mathcal{S}_{\Psi^{\text{Timediv}}}(\langle s_{\tau}, s_{\mathfrak{s}} \rangle)$ 
1  $\text{Reg}(\llbracket \alpha \cdot \mathcal{T}_{\tau} \rrbracket), \text{Reg}(\llbracket \alpha \cdot \mathcal{T}_{\mathfrak{s}} \rrbracket) :=$  Region graphs of the expanded timed game structures
    $\llbracket \alpha \cdot \mathcal{T}_{\tau} \rrbracket$  and  $\llbracket \alpha \cdot \mathcal{T}_{\mathfrak{s}} \rrbracket$ ;
2  $\mathfrak{F} := \mathfrak{F}(\text{Reg}(\llbracket \alpha \cdot \mathcal{T}_{\tau} \rrbracket), \text{Reg}(\llbracket \alpha \cdot \mathcal{T}_{\mathfrak{s}} \rrbracket));$ 
   /* Finite weighted game constructed from the region graphs */
3 switch  $\Psi$  do
4   | case  $\mathcal{D}_{\text{MaxDiff}}$ 
5   |    $\Xi := \text{DiffSumCB}_{\text{coBüchi}(ticks_{\tau}=0)};$ 
6   | case  $\mathcal{D}_{\text{LimMaxDiff}}$ 
7   |    $\Xi := \text{EvDiffSumCB}_{\text{coBüchi}(ticks_{\tau}=0)};$ 
8   | case  $\mathcal{D}_{\text{AvgDiff}}$ 
9   |    $\Xi := \text{AvDiffSumCB}_{\text{coBüchi}(ticks_{\tau}=0)};$ 
10  |
11 endsw
12 return  $\alpha^{-1} \cdot \text{Opt}^{\mathfrak{F}}(\Xi) \left( \left\langle \begin{array}{c} \text{Reg}(\llbracket \alpha \cdot s_{\tau} \rrbracket), \\ \text{Reg}(\llbracket \alpha \cdot s_{\mathfrak{s}} \rrbracket), \\ 2 \end{array} \right\rangle \right);$ 

```

Function $h_{\Psi, \alpha}(s_{\tau}, s_{\mathfrak{s}})$

- [CD10] K. Chatterjee and L. Doyen. “Energy Parity Games”. In: *ICALP*. LNCS 6199. Springer, 2010, pp. 599–610.
- [Cer92] K. Cerans. “Decidability of Bisimulation Equivalences for Parallel Timer Processes.” In: *CAV*. LNCS 663. Springer, 1992, pp. 302–315.
- [CGL12] S. Chaudhuri, S. Gulwani, and R. Lublinerman. “Continuity and robustness of programs”. In: *Commun. ACM* 55.8 (2012), pp. 107–115.
- [Cha+10] K. Chatterjee et al. “Generalized Mean-payoff and Energy Games”. In: *FSTTCS*. LIPIcs 8. Schloss Dagstuhl - LZI, 2010, pp. 505–516.
- [Cha10] J. Chaloupka. “Z-Reachability Problem for Games on 2-Dimensional Vector Addition Systems with States Is in P”. In: *RP*. 2010, pp. 104–119.
- [CHJ05] K. Chatterjee, T.A. Henzinger, and M. Jurdzinski. “Mean-Payoff Parity Games”. In: *LICS*. IEEE Computer Society, 2005, pp. 178–187.
- [Des+04] J. Desharnais et al. “Metrics for labelled Markov processes”. In: *Theor. Comput. Sci.* 318.3 (2004), pp. 323–354.
- [EJ91] E.A. Emerson and C. Jutla. “Tree automata, mu-calculus and determinacy”. In: *FOCS’91*. IEEE, 1991, pp. 368–377.
- [GJP08] A. Girard, A. A. Julius, and G. Pappas. “Approximate simulation relations for hybrid systems.” In: *Discrete Event dynamic Systems* 18.2 (2008), pp. 163–179.
- [GPT10] A. Girard, G. Pola, and P. Tabuada. “Approximately Bisimilar Symbolic Models for Incrementally Stable Switched Systems”. In: *Automatic Control, IEEE Transactions on* 55.1 (2010), pp. 116–126.
- [HMP05] T. A. Henzinger, R. Majumdar, and V. S. Prabhu. “Quantifying similarities between timed systems”. In: *FORMATS*. LNCS 3829. Springer, 2005, pp. 226–241.
- [JS03] J. Jacod and A. Shiryaev. *Limit Theorems for Stochastic Processes*. Springer, 2003.
- [Pol+10] G. Pola et al. “Symbolic models for nonlinear time-delay systems using approximate bisimulations”. In: *Systems & Control Letters* 59.6 (2010), pp. 365–373.
- [QFD11] J.D. Quesel, M. Fränzle, and W. Damm. “Crossing the Bridge between Similar Games”. In: *FORMATS 2011*. LNCS 6919. 2011, pp. 160–176.
- [VR11] Y. Velner and A. Rabinovich. “Church Synthesis Problem for Noisy Input”. In: *FOSSACS*. 2011, pp. 275–289.